



Année Universitaire 2023-2024
MÉMOIRE DE STAGE

MÉMOIRE DE STAGE



Présenté par

Esteban ELIAS PUEYO

Jury

IUT : M. MARTIN

IUT : Mme CHOLLAT-NAMY

Société : M. SCHWAB

Déclaration de respect des droits d'auteurs

Par la présente, je déclare être le seul auteur de ce rapport et assure qu'aucune autre ressource que celles indiquées n'ont été utilisées pour la réalisation de ce travail. Tout emprunt (citation ou référence) littéral ou non à des documents publiés ou inédits est référencé comme tel.

Je suis informé qu'en cas de flagrant délit de fraude, les sanctions prévues dans le règlement des études en cas de fraude aux examens par application du décret 92-657 du 13 juillet 1992 peuvent s'appliquer. Elles seront décidées par la commission disciplinaire de l'UGA.

A Grenoble, le 24 juin 2024

Remerciements

Je souhaite tout d'abord exprimer ma gratitude à M. Schwab pour m'avoir offert cette opportunité de stage. Je tiens également à remercier M. Arrigo pour son soutien constant et l'aide précieuse qu'il m'a apportée tout au long de la durée du stage.

Sommaire

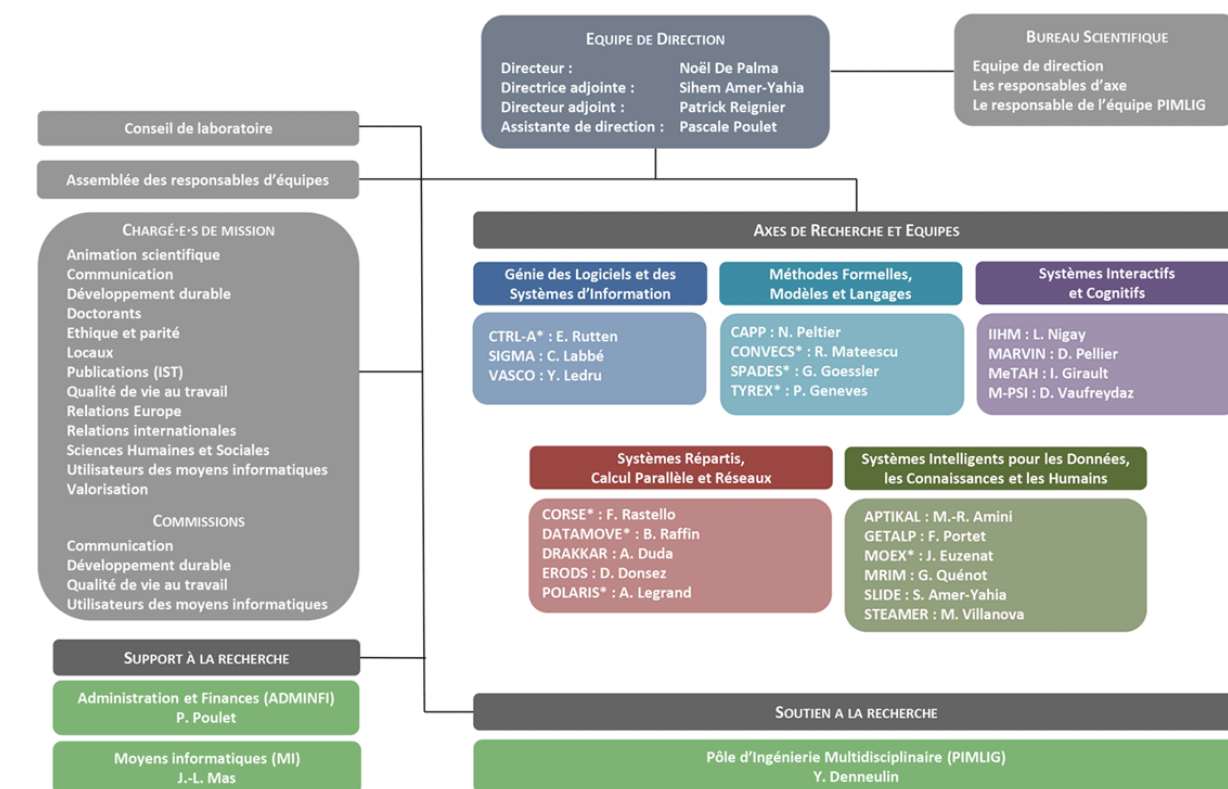
Table des matières

Résumé long en français	4
I. Introduction	6
<i>I.1 Présentation du LIG</i>	6
<i>I.2 Cadre de travail</i>	7
<i>I.3 Analyse de l'existant</i>	8
<i>I.4 Présentation de la mission</i>	8
II. Missions	9
<i>II.1 Technologies utilisées</i>	9
<i>II.2 Prise en main de AugCom</i>	9
<i>II.3 Ajout de l'export vers Word</i>	10
<i>II.4 Ajouts des boutons sons et vidéos</i>	11
<i>II.5 Corrections des bugs</i>	12
III. Conclusion	13
Glossaire	14
Annexes	15
Résumé :	16
Abstract :	17

I. Introduction

I.1 Présentation du LIG

Le stage s'est produit au Laboratoire Informatique de Grenoble (LIG). Il a été créé le 1er janvier 2007. D'après son livret de 2021, le LIG regroupe environ 450 chercheurs, enseignants-chercheurs, doctorants et personnels en soutien à la recherche. Il est de plus en partenariat avec l'Université Grenoble-Alpes (UGA), l'INP (Institut Polytechnique) de Grenoble, le Centre National de la Recherche Scientifique (CNRS) et l'Institut National de Recherche en Informatique et en Automatique (INRIA). Le laboratoire est étendu sur trois sites : un sur le campus, au sein du bâtiment IMAG (Institut d'informatique et mathématiques appliquées de Grenoble), un sur le campus Minatec et un autre à Montbonnot. Le LIG se veut un acteur actif à la recherche et au développement des différents domaines de l'informatique.



* Equipe projet commune INRIA

UMR5217 – www.liglab.fr – septembre 2023

Figure 1 : Organigramme LIG

Le laboratoire compte au total 22 équipes de recherche, chacune portant sur des projets et des recherches spécifiques répartis selon cinq axes thématiques : Génie des logiciels et des systèmes d'informations ; Méthodes formelles, modèles et langages ; Systèmes intelligents pour les données, les connaissances et les humains ; Systèmes interactifs et cognitifs ; Systèmes répartis, calcul parallèle et réseaux.

Dans le cadre de mon stage, j'ai été intégré au Groupe d'Étude pour la Traduction Automatique et le Traitement Automatisé des Langues et de la Parole (GETALP). Cette équipe mène des recherches sur les systèmes intelligents pour les données, les connaissances et les humains, avec un accent particulier sur le développement de la communication et du traitement de l'information multilingue.

Un des projets de cette équipe est InterAACtionBox, un dispositif destiné à favoriser l'apprentissage et la communication des personnes en situation de handicap cognitif. Ce projet intègre un ordinateur ou une tablette permettant des interactions oculaires et tactiles, un système d'exploitation spécifique qui gère InterAACtionBox, ainsi que plusieurs logiciels de Communication Alternative et Augmentée (CAA). Ces outils sont conçus pour offrir des solutions de communication adaptées aux besoins des utilisateurs tout en restant à un prix abordable. Ce projet est également dirigé par mon tuteur, M. Didier Schwab.

1.2 Cadre de travail

Mon stage s'est déroulé dans l'un des bâtiments principaux du Laboratoire d'Informatique de Grenoble (LIG), plus précisément dans le bâtiment IMAG. J'étais installé dans un bureau partagé avec trois autres stagiaires. Un mois après le début de mon stage, deux autres stagiaires issus du BUT nous ont rejoints, augmentant ainsi notre équipe de travail.

Initialement, je travaillais sur ma machine personnelle. Cependant, celle-ci a cessé de fonctionner en cours de stage, me contraignant à solliciter un ordinateur fixe auprès du LIG. Durant toute la durée de mon stage, j'ai exclusivement utilisé Linux comme système d'exploitation. Sur ma machine personnelle, j'étais sous Ubuntu 22.04. Après le passage à l'ordinateur fixe prêté par le LIG, j'ai continué à utiliser Linux, cette fois avec l'environnement de bureau Gnome 43.9.

Durant mon stage, mes horaires étaient flexibles, ce qui me permettait d'organiser ma journée de travail selon mes préférences, tout en respectant certaines contraintes. Je devais arriver au bureau entre 8h et 10h et assurer un total de 35 heures de travail par semaine. Bien que je devais être présent chaque matin et chaque après-midi, j'avais la liberté de répartir mes heures de travail de manière à convenir à mon emploi du temps personnel.

Pour signaler ma présence, il m'était demandé d'envoyer un message sur Discord à mon heure d'arrivée et un autre au moment où je quittais le bâtiment du LIG. Je devais également envoyer un message pour signaler la fin de ma journée et les tâches que j'avais effectué durant cette dernière.

1.3 Technologies utilisées

Angular CLI

Angular CLI (Command Line Interface) est un framework open source développé par Google pour la création d'applications web. Basé sur TypeScript, un langage de programmation orienté objet qui étend JavaScript en ajoutant des fonctionnalités supplémentaires, Angular utilise une approche de développement par composants. Cela permet de construire des applications en blocs modulaires et réutilisables.

Le principal avantage d'Angular est sa capacité à créer des applications à page unique (SPA), où une seule page est chargée initialement et le contenu est mis à jour dynamiquement sans recharger la page entière. Cette approche offre une expérience utilisateur plus rapide et fluide.

Angular adopte un modèle de développement basé sur le MVC (Modèle - Vue - Contrôleur), où les composants servent à la fois de vues et de contrôleurs. Les composants, blocs fondamentaux d'une application Angular, sont responsables de l'affichage des données et de la gestion des interactions utilisateur. En outre, Angular propose un système de liaison de données bidirectionnelle, ce qui signifie que les modifications apportées aux données dans le code sont automatiquement reflétées dans l'interface utilisateur et vice versa.

Le framework Angular est également doté de multiples fonctionnalités qui simplifient le développement. Parmi celles-ci, on trouve la gestion des formulaires, la validation des données, la gestion des routes, la communication avec les API backend, la gestion des événements et la mise en place de gardes. Ces fonctionnalités avancées permettent aux développeurs de gagner en productivité et de créer des applications robustes et performantes.

Cependant, l'application sur laquelle je travaille utilise la version 9.1.15 d'Angular, qui est totalement obsolète étant donné que nous en sommes actuellement à la version 18 (22 Mai 2024). Cette version ne bénéficie plus de mises à jour de sécurité ni de nouvelles fonctionnalités. Son obsolescence me pose également des problèmes lorsque je souhaite utiliser des bibliothèques pour certaines fonctionnalités ou lorsque je fais des recherches sur la technologie.

Node JS

Node.js est un environnement d'exécution JavaScript open-source et multiplateforme qui permet d'exécuter du code JavaScript côté serveur. Il utilise le moteur V8 de Google Chrome pour exécuter le JavaScript de manière efficace et rapide. Node.js est basé sur un modèle événementiel et non bloquant, ce qui le rend idéal pour des applications en temps réel à grande échelle et pour les systèmes nécessitant de nombreuses opérations d'E/S.

Node.js bénéficie d'un écosystème riche de modules grâce à npm (Node Package Manager), ce qui facilite l'ajout de fonctionnalités à une application. Il est principalement utilisé pour construire des serveurs web, des API RESTful, et des applications en temps réel telles que les chats et les jeux en ligne. Sa capacité à gérer de nombreuses connexions simultanément avec une grande efficacité est l'une de ses principales forces.

Toutefois, comme pour Angular, l'obsolescence de la version utilisée a posé des défis. Dans ce projet, nous utilisons la version 14.17.6 de Node.js, qui est maintenant obsolète. Cette ancienne version a entraîné des difficultés similaires à celles rencontrées avec Angular, notamment en termes de compatibilité avec les modules les plus récents et de disponibilité de la documentation à jour.

GitHub

GitHub est une plateforme de développement collaboratif basée sur Git, un système de contrôle de version distribué. Fondée en 2008, GitHub permet aux développeurs de gérer et de partager leur code, de suivre les modifications apportées à leurs projets et de collaborer efficacement avec d'autres développeurs. Grâce à ses fonctionnalités robustes de gestion de version, les équipes peuvent travailler simultanément sur le même projet sans conflit, tout en conservant un historique complet des modifications.

Oltre le contrôle de version, GitHub propose des outils de collaboration comme les pull requests, qui permettent aux développeurs de proposer des modifications et de discuter de leur intégration dans le projet principal. Les issues et le système de suivi des bugs aident à gérer les tâches et les problèmes, améliorant ainsi la gestion de projet. GitHub Actions offre des capacités d'intégration continue et de déploiement continu (CI/CD), automatisant les workflows de développement et facilitant la livraison de code de haute qualité.

Pendant toute la durée de mon stage, j'ai utilisé GitHub. La première étape consistait à forker le projet AugCom à partir de leur dépôt GitHub. Cette pratique visait à réduire les risques d'erreurs lors de mes push. En gérant mon propre dépôt GitHub, j'ai pu travailler de manière autonome et sécurisée. Après avoir accompli chaque tâche majeure, j'avais la responsabilité de créer une pull request. Cette requête était ensuite soumise à l'examen de Jordan Arrigo, qui décidait de l'accepter ou de la refuser après avoir éventuellement proposé des modifications à apporter.

Discord

Discord permet de créer des serveurs regroupant les membres du projet, et facilite grandement la communication écrite et le partage de petits documents tels que des images ou des liens. Le projet InterAAction étant de plus un projet open-source ne traitant pas de données sensibles, l'utilisation de Discord ne pose pas de problème à l'intégrité du projet. Enfin, Discord permet une communication rapide et efficace, adaptée à la résolution de problèmes et à l'entraide entre les membres du projet.

I.4 Présentation de la mission

Durant mon stage, mes missions se sont divisées en deux parties principales : l'ajout de nouvelles fonctionnalités et la réparation des bugs existants.

Une part importante de mon travail a consisté à identifier et corriger les bugs présents dans l'application. Cette tâche nécessitait une analyse minutieuse du code pour localiser les erreurs et comprendre leurs origines. Une fois les bugs identifiés, j'ai élaboré des solutions pour les corriger, tout en veillant à ne pas introduire de nouveaux problèmes. Ce processus impliquait souvent de tester rigoureusement les correctifs dans différents scénarios pour s'assurer qu'ils fonctionnaient correctement. La documentation des bugs et des solutions apportées faisait également partie intégrante de ce travail, afin de faciliter la compréhension et la résolution de problèmes similaires à l'avenir.

En parallèle de la correction des bugs, j'étais responsable du développement de nouvelles fonctionnalités pour l'application, et j'avais la liberté totale de choisir la manière dont je les implémenter. Cette autonomie m'a permis d'explorer différentes approches et technologies pour concevoir des solutions innovantes. Mon travail commençait par une phase de planification, où je définissais les objectifs et les spécifications de chaque nouvelle fonctionnalité en fonction des besoins des utilisateurs et des directives générales du projet.

Ensuite, je passais à la phase de développement, où je concevais et codais les nouvelles fonctionnalités en veillant à leur intégration harmonieuse avec l'architecture existante de l'application. Cette phase comprenait également l'écriture de tests unitaires et fonctionnels pour m'assurer que les nouvelles fonctionnalités étaient robustes et fiables. Chaque nouvelle fonctionnalité devait non seulement répondre aux exigences initiales, mais aussi fonctionner sans compromettre la stabilité et la performance de l'application.

Enfin, après avoir terminé le développement, je réalisais une série de tests approfondis pour vérifier la compatibilité des nouvelles fonctionnalités avec les autres parties de l'application. Je documentais également chaque nouvelle fonctionnalité, fournissant des instructions détaillées sur son utilisation et son intégration, afin de faciliter la maintenance future et l'adoption par les utilisateurs. Cette expérience m'a permis de développer des compétences précieuses en gestion de projet et en développement de logiciels, tout en contribuant de manière significative à l'amélioration de l'application.

1.5 Analyse de l'existant

Ma mission dans ce projet consiste à ajouter des fonctionnalités et à réparer des bugs sur des applications existantes. On m'a introduit à 3 applications bien distinctes.

InterAActionPlayer

InterAAction Player est une application dont l'objectif est de permettre aux personnes en situation de handicap d'accéder aux plateformes vidéo et musique Youtube, Deezer, et Spotify de manière oculaire. Développé en Angular, l'application web possède une page principale permettant de manipuler les médias ajoutés (jouer, pauser, modifier le volume sonore, mettre en plein écran, etc.), une page permettant d'éditer les options (thème, taille des boutons, activation/désactivation de la sélection par fixation), et une dernière page permettant de rechercher d'autres médias à ajouter. Un système d'utilisateurs et la sauvegarde de leurs options et playlists est aussi inclus dans l'application.

Figure 2 : Page Principale InterAActionPlayer

InterAActionScene

InterAActionScene est une application permettant à l'utilisateur de créer des scènes, de leurs attribuer des images et d'y placer des dessins ou des "hotspots" de manière oculaire. Elle possède également une page pour les options. L'application est codée en Angular, cependant, je n'ai pas eu l'opportunité de travailler sur cette application.

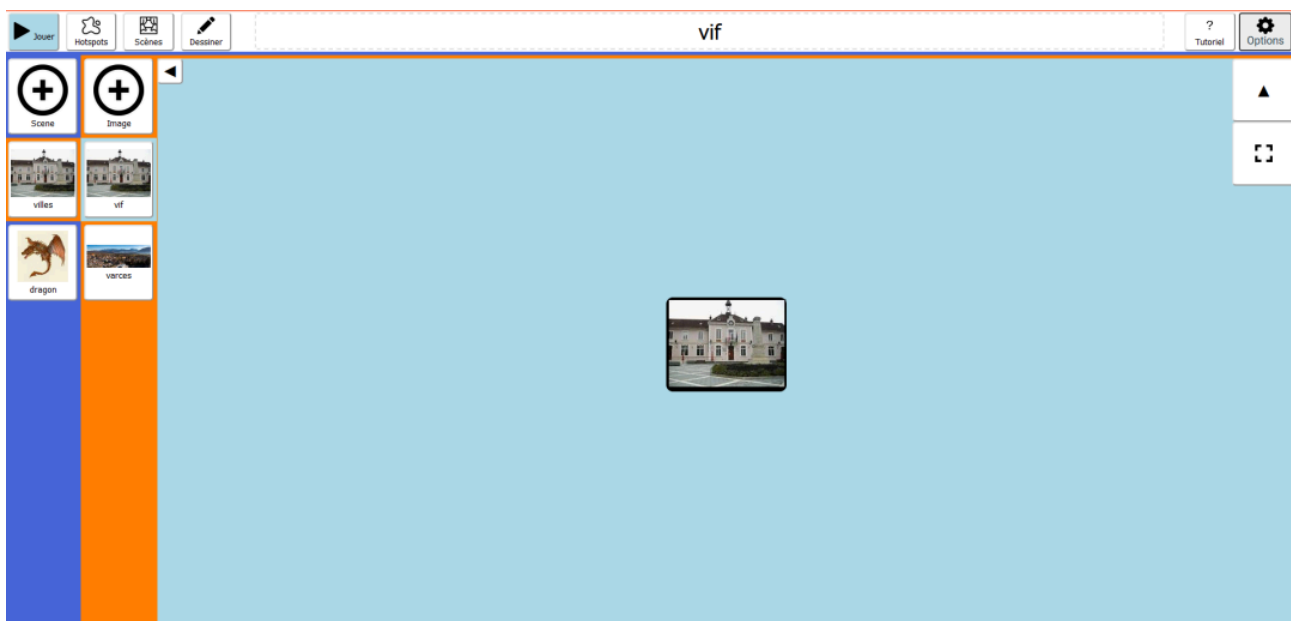


Figure 3 : Page Principale InterAACtionScene

AugCom

AugCom est une application permettant la création de grilles de communication alliant pictogramme et mots. Cette application aide les personnes en situation de handicap cognitif à communiquer en leur permettant de sélectionner les mots souhaités sur la grille et de les prononcer à haute voix pour eux.

AugCom est l'application la plus complexe parmi les trois. Elle est développée en Angular et adopte une architecture classique propre à ce framework. AugCom est constituée d'environ soixante composants, chacun ayant une fonction spécifique au sein de l'application. En outre, elle intègre une vingtaine de services qui facilitent la gestion des données et la communication entre les différentes parties de l'application. Tous ces éléments sont répartis sur une quinzaine de pages.

La page principale d'AugCom intègre la fonctionnalité centrale de l'application : la prononciation des mots affichés dans les grilles. Pour utiliser cette fonctionnalité, l'utilisateur clique sur un bouton associé à un mot spécifique. Le mot est d'abord prononcé à voix haute, puis il est ajouté à la barre supérieure de l'écran. Une fois plusieurs mots sélectionnés, ils peuvent être combinés pour former une phrase complète, qui peut ensuite être prononcée en entier en utilisant le bouton de lecture.

Cette page principale offre également un accès à d'autres fonctionnalités essentielles. L'utilisateur peut éditer les grilles et les boutons, ce qui permet de personnaliser les options disponibles en fonction de ses besoins. De plus, il y a un accès aux paramètres de l'application, où l'utilisateur peut ajuster divers réglages pour optimiser son expérience. Enfin, un tutoriel est disponible pour guider

les nouveaux utilisateurs à travers les fonctionnalités de l'application. J'ai été chargé de retravailler ce tutoriel pour le rendre plus clair et plus accessible, assurant ainsi une meilleure compréhension et utilisation de l'application par tous les utilisateurs.

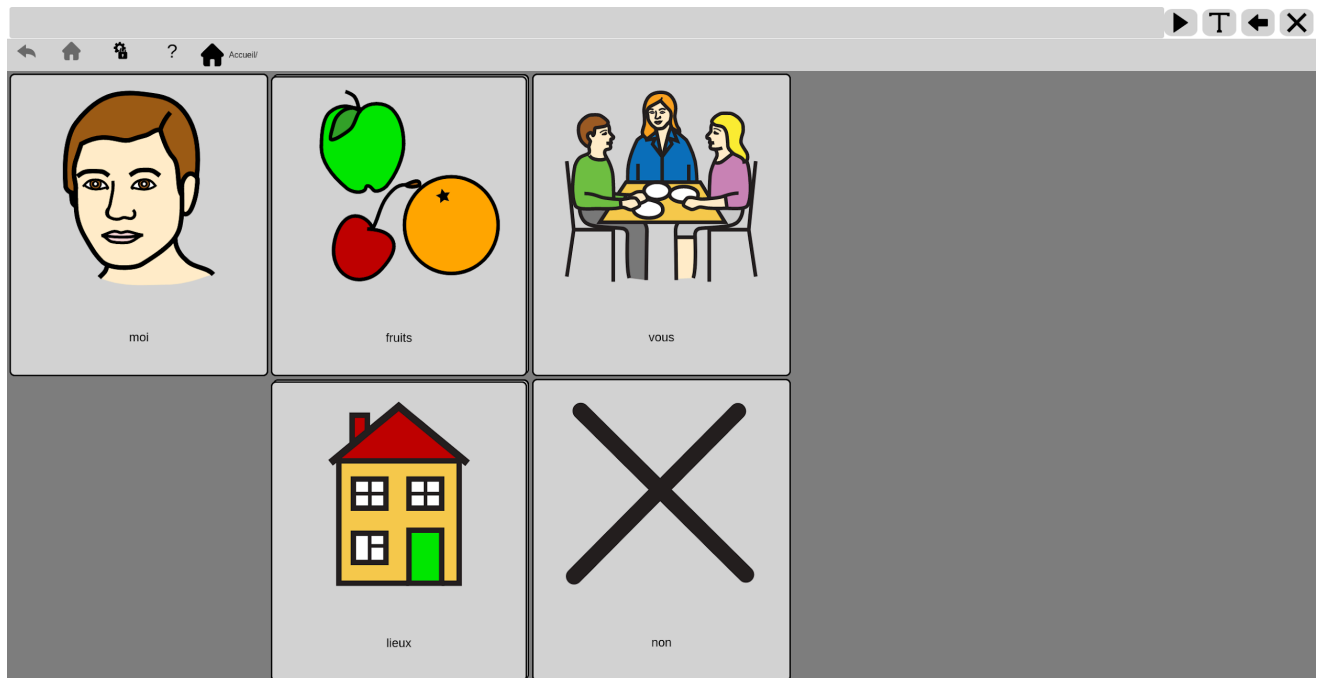



Figure 2 : Page Principale d'AugCom

Grâce à un appui sur le bouton en forme de crayon, nous pouvons entrer dans le mode édition de l'application. Depuis ce mode, il est possible de modifier les grilles selon nos besoins et d'accéder au mode édition des boutons. Toutefois, pour qu'une grille soit enregistrée, il est nécessaire de la télécharger sur votre machine. À chaque utilisation du site, il faudra également charger la grille téléchargée pour pouvoir l'utiliser.

Annexe : Mode Edition

En double-cliquant sur un bouton dans le mode édition, nous accédons à la page dédiée à l'édition des boutons. Cette page est organisée en quatre sections distinctes pour faciliter la personnalisation complète des boutons. La première section permet de choisir le type de bouton. La deuxième section est consacrée à l'apparence du bouton. L'utilisateur peut y personnaliser le pictogramme du bouton, choisir la couleur de fond et définir la couleur du contour. La troisième section offre des options pour définir des formes alternatives du bouton. Cette section est là pour les mots pour intégrer des synonymes de mots ou des images alternatives à la principale choisie. Enfin, la quatrième section concerne les interactions possibles avec le bouton. L'utilisateur peut configurer les actions qui se déclenchent lors d'un simple clic, d'un appui long ou d'un double-clic sur le bouton. J'ai notamment dû travailler sur cette page durant mon stage.

Retour
AugCom



Informations
Apparence
Formes Alternatives
Interactions

Enregistrer

Modifier les informations:

Nom de l'élément :
Type de l'élément :

vous

@Bouton
L'élément créé sera un simple bouton
CBouton son
L'élément créé sera un bouton qui lancera un bruit quand il sera utilisé
CBouton video
L'élément créé sera un bouton qui lancera une vidéo quand il sera utilisé
CDossier
L'élément créé sera un dossier pouvant contenir d'autres boutons

AugCom-AFSR 1.5.1 Dev v.2022-09-12

Figure : Page d'édition des boutons

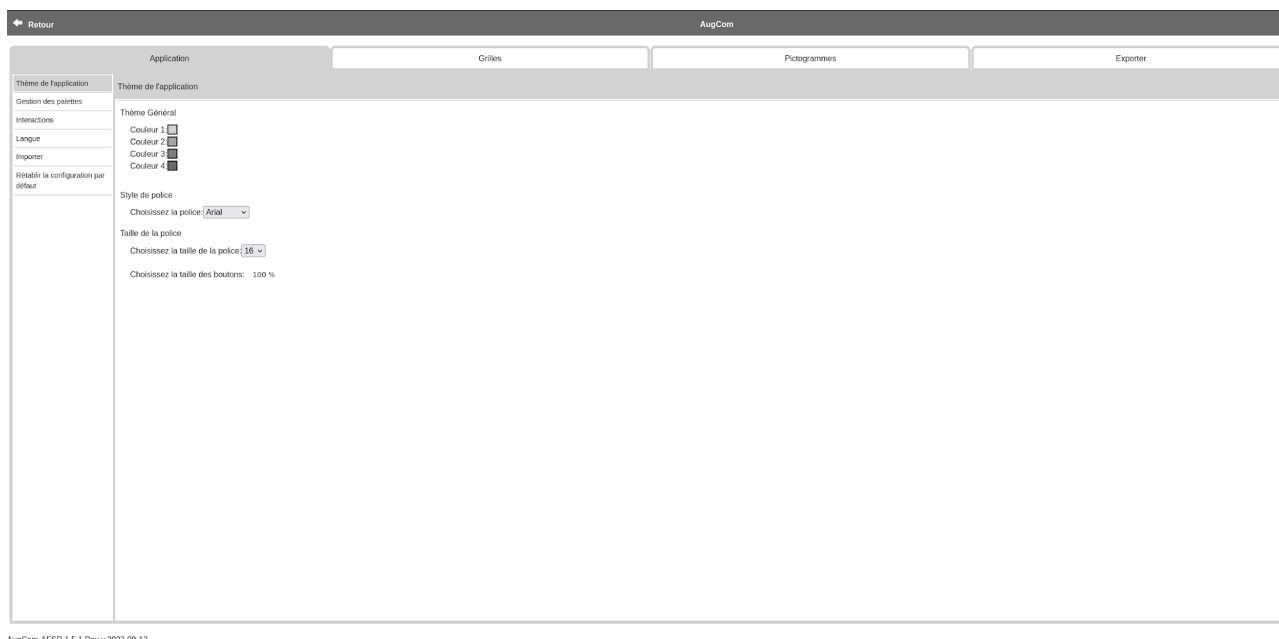
La page des paramètres, accessible depuis la page principale de l'application, est divisée en quatre grandes sous-parties, chacune offrant des options de personnalisation et de gestion avancées.

La première sous-partie permet de changer le thème de l'application. Les utilisateurs peuvent ajuster les couleurs du thème de l'application qui sont laissées très simpliste de base, les polices et d'autres éléments visuels, afin de rendre l'interface plus agréable et adaptée à leurs préférences.

La deuxième sous-partie concerne la gestion des grilles. Elle offre des outils pour créer de nouvelles grilles, modifier les grilles existantes, et organiser les différentes grilles en fonction des besoins de l'utilisateur.

La troisième sous-partie est dédiée à une meilleure gestion des pictogrammes. Les utilisateurs peuvent modifier la position des pictogrammes sur les boutons et leur taille. Cette sous partie possède également des mentions de droit d'auteurs pour les pictogrammes utilisées dans l'application.

Enfin, la quatrième sous-partie permet l'exportation des grilles. Les utilisateurs peuvent exporter leurs grilles sous différents formats, notamment PDF, Word ou Excel, facilitant ainsi le partage et l'impression des grilles pour une utilisation hors ligne ou pour une documentation plus formelle.



AugCom-AFSR 1.5.1 Dev v.2022.09.12

Figure : Page Paramètre

L'application est disponible en deux langues : anglais et français. Les traductions sont stockées dans un fichier JSON. Chaque phrase est enregistrée sous sa version anglaise et sa version française, avec un identifiant unique pour chaque entrée. Ce fichier JSON est ensuite utilisé par un service dédié qui, en fonction de la langue sélectionnée par l'utilisateur, récupère et affiche la version appropriée du texte.

II. Missions

II.1 *Prise en main de AugCom*

En attendant que l'on me trouve des missions concrètes parmi les issues d'AugCom j'ai décidé de me lancer dans une refonte du guide de l'application. Cela me permettait de me familiariser avec les différentes fonctionnalités d'AugCom et d'étudier son code au passage, en effet, le guide d'AugCom n'est pas un PDF mais une page web intégré à l'application. De plus, ce dernier n'avait pas été mis à jour depuis un certain temps et était devenu obsolète sur plusieurs points.

Dans un premier temps, j'ai donc étudié et testé les différentes fonctionnalités que le guide actuel présentait. Cette étape m'a permis de me familiariser avec les aspects existants de l'application et de comprendre comment chaque fonctionnalité est censée fonctionner.

Ensuite, j'ai dressé une liste des fonctionnalités qui avaient évolué par rapport à l'ancien guide. Une fois cette analyse terminée, j'ai commencé à rédiger les nouvelles sections du guide, en mettant à jour les descriptions des fonctionnalités qui avaient changé. J'ai pris soin de fournir des explications claires et détaillées pour chaque point, afin de garantir que les utilisateurs comprennent bien les nouvelles fonctionnalités et leur utilisation. Par ailleurs, j'ai pris de nouvelles captures d'écran de toutes les fonctionnalités présentées dans le guide.

Figure : Guide AugCom

Cependant, comme mentionné précédemment, AugCom est disponible en deux langues : français et anglais. Par conséquent, j'ai dû traduire le guide en anglais pour garantir son accessibilité à tous les utilisateurs anglophones. En plus de la traduction, j'ai également pris des captures d'écran de la version anglaise des fonctionnalités.

L'une des difficultés que j'ai rencontrées lors de la mise à jour du guide a été de comprendre comment il était intégré à l'application. Le guide apparaît sous forme de pop-up lorsqu'on accède aux paramètres de l'application. Toutefois, en explorant le composant des paramètres, je ne trouvais aucune indication concernant le guide. Cette absence d'information dans le composant des paramètres m'a initialement déconcerté.

Cependant, en analysant plus en détail l'architecture de l'application, j'ai découvert que le guide possédait son propre composant distinct. En comprenant cela, j'ai pu effectuer les mises à jour

nécessaires de manière plus efficace et précise, en intégrant les nouvelles sections et les traductions sans problème via l'HTML.

La deuxième difficulté que j'ai rencontrée a été de comprendre comment le site gère les traductions. En effet, AugCom est disponible en anglais et en français, ce qui nécessite une gestion efficace des langues. Il m'a fallu du temps pour saisir le fonctionnement du système de traduction, car toutes les traductions sont stockées dans un fichier JSON appelé par un service spécifique.

Chaque phrase dans l'application possède un identifiant unique. Cet identifiant est ensuite utilisé dans une fonction incluse dans le HTML pour afficher la traduction appropriée en fonction de la langue sélectionnée par l'utilisateur. Comprendre cette structure a été un défi, car il fallait non seulement repérer où et comment les identifiants étaient définis, mais aussi comprendre le mécanisme par lequel le service récupérait et injectait les traductions correctes dans l'interface utilisateur.

```
1 {  
2   "dictionary" : [  
3     { "id" : "back" , "FR" : "Retour", "EN" : "Back" },  
4     { "id" : "modifInfo" , "FR" : "Modifier les informations:", "EN" : "Edit informations:" },  
5     { "id" : "soon" , "FR" : "A venir...", "EN" : "Available soon..." },  
6     { "id" : "settings" , "FR" : "Paramètres", "EN" : "Settings" },  
7     { "id" : "account" , "FR" : "Compte", "EN" : "Account" },  
8     { "id" : "accountInfo" , "FR" : "Informations du compte", "EN" : "Account Informations" },  
9     { "id" : "saveManagement" , "FR" : "Gestion des sauvegardes", "EN" : "Save Management" },  
10    { "id" : "appearance" , "FR" : "Apparence", "EN" : "Appearance" },  
11    { "id" : "globalAppearance" , "FR" : "Apparence générale", "EN" : "Global Appearance" },  
12    { "id" : "paletteManagement" , "FR" : "Gestion des palettes", "EN" : "Palette Management" },
```

Figure 3 : Capture d'écran fichier json

Une fois que j'ai maîtrisé ce système, j'ai pu modifier les traductions existantes et ajouter de nouvelles phrases avec leurs correspondances en anglais et en français. J'ai également mis à jour le fichier JSON et vérifié que chaque modification était correctement reflétée dans l'application.

II.2 Ajout de l'export vers Word

Explication de la mission

La deuxième tâche que j'ai entreprise a été de coder une méthode pour exporter automatiquement la grille de l'application vers un document Word. Comme mentionné dans l'analyse de l'existant, AugCom possède déjà des moyens d'exporter la grille vers PDF et Excel. Cependant, afin d'étendre les possibilités d'export il a été décidé de rajouter Word comme moyen d'export.

Développement de la mission

Pour commencer, j'ai examiné la fonctionnalité d'exportation existante vers Excel, qui utilise la librairie « xlsx ». Ma première idée a été de vérifier si cette librairie pouvait être adaptée pour un export vers Word. Cependant, après une analyse approfondie, j'ai constaté qu'elle n'était pas adaptée à cet usage.

Je me suis donc mis à rechercher une librairie appropriée pour l'exportation vers Word. À ce moment-là, j'ai rencontré un obstacle lié à l'obsolescence des technologies utilisées dans l'application. Bien que de nombreuses librairies existent pour les dernières versions d'Angular, il y avait très peu de documentation et de librairies compatibles avec la version de Angular utilisée par AugCom. Après plusieurs jours de recherche, j'ai trouvé la librairie « docx » qui semblait convenir.

J'ai alors codé la fonction d'exportation et procédé aux premiers tests.

Cependant, La librairie causait des erreurs de compilation dues à des incompatibilités. Après de nombreux essais pour résoudre ce problème, j'ai décidé de demander de l'aide à Jordan Arrigo. Sous ses conseils, j'ai downgradé la librairie version par version jusqu'à trouver une version fonctionnelle. Néanmoins, un nouveau problème est apparu : l'utilisation de « fs file reader » pour inclure des images ne fonctionnait pas avec Angular.

Pour contourner ce problème, j'ai opté pour une solution de conversion des images en Base64. Bien que cette méthode ait résolu le problème initial, elle a présenté une nouvelle limitation : les documents Word générés étaient compatibles avec LibreOffice et Microsoft Word, mais Google Docs ne pouvait pas afficher les images correctement. Cette incompatibilité a été notée et ajoutée comme une issue à résoudre plus tard dans le GitHub d'AugCom.

Enfin, l'export initial n'était pas complet et j'ai dû le retravailler pour prendre en compte les boutons "folder" de l'application, qui peuvent contenir d'autres boutons. J'ai utilisé du codage récursif pour inclure ces boutons correctement dans le document Word exporté. Après plusieurs tests et ajustements, j'ai présenté mes résultats à Jordan Arrigo pour validation. Après avoir affiné la fonctionnalité selon ses retours, j'ai soumis une pull request. Mon code a été testé et approuvé, bien que l'issue concernant l'affichage des images dans Google Docs reste à résoudre.

Voici une capture d'écran du rendu final de l'export du document Word. Sur ce document, les éléments "fruits" et "lieux" sont des boutons "folders" qui génèrent des sous-dossiers.

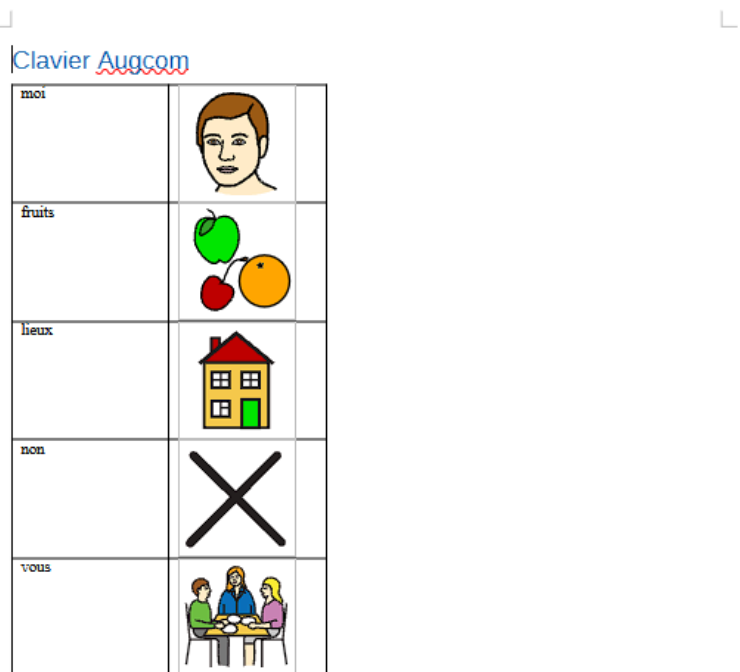


Figure : Capture d'écran du rendu final du clavier

Annexe : Capture d'écran du rendu final sous dossier "fruits"

II.3 Ajouts des boutons sons et vidéos

Explication de la mission

Une des autres tâches qui m'a été confiée consistait à développer deux nouveaux types de boutons pour l'application AugCom. Comme mentionné précédemment, AugCom est une grille interactive qui utilise des boutons pour composer des phrases. Je devais donc rajouter l'option de créer des boutons sons et vidéos.

Pour les boutons sonores, l'objectif était de permettre aux utilisateurs de sélectionner un fichier audio qui serait joué immédiatement après avoir cliqué sur le bouton. Cela pourrait être utilisé, par exemple, pour intégrer des enregistrements vocaux ou des effets sonores.

Quant aux boutons vidéo, il s'agissait de permettre l'intégration de clips vidéo qui se lanceraient dès que l'utilisateur appuierait sur le bouton correspondant. Ces clips devaient pouvoir provenir soit de la machine personnelle de l'utilisateur, soit du web, comme par exemple une vidéo YouTube.

Développement de la mission

Pour commencer, j'ai étudié la manière dont les grilles sont construites dans l'application, puis j'ai analysé plus en détail comment les boutons sont gérés dans le code. Les boutons ne sont pas enregistrés dans une base de données ou un fichier ; ce ne sont que des objets volatils dans l'application, ce qui complique leur gestion et leur modification. En réalité, la seule distinction entre les différents types de boutons repose sur un attribut de type chaîne de caractères. Pour les boutons de type dossier, cet attribut est initialisé à "folder", tandis que pour les boutons normaux, il est initialisé à "button".

J'ai donc simplement rajouté la possibilité de créer les boutons sons et vidéos, en définissant cet attribut à « sound » pour les boutons sons et à « video » pour les boutons vidéos. Ensuite, je me suis concentré sur la création de ces boutons via le frontend. J'ai ajouté des options au questionnaire utilisé pour créer un bouton, incluant désormais les options « sound » et « video » avec une explication de l'utilisation de ces deux types de boutons.

Si l'utilisateur choisissait de créer un bouton « sound », il aurait alors l'option d'ajouter un fichier audio enregistré sur sa machine. Pour les boutons « video », l'utilisateur pourrait sélectionner un fichier vidéo. J'ai configuré le formulaire pour qu'il n'accepte que les fichiers audio pour les boutons sonores et les fichiers vidéo pour les boutons vidéo.

Cependant, j'ai dû ajouter un attribut aux boutons pour stocker le son ou la vidéo, ce qui a considérablement compliqué le codage de la fonctionnalité. L'ajout de cet attribut nécessitait de modifier chaque partie du code où des boutons étaient créés. Ces boutons étaient appelés dans des classes pour être utilisés par des outils extérieurs à AugCom, tels que LifeCompagnon. Pour intégrer cet attribut, j'ai dû modifier toutes les instances de création de boutons dans le code.

Malgré ces efforts, j'ai dû m'arrêter au moment où le son devait être joué. Je compte faire de mon mieux pour terminer cette fonctionnalité dans la semaine restante. Cependant, la partie majoritaire

du travail est faite. Nous pouvons voir sur cette capture d'écran de la page de modification des boutons que les boutons vidéo et son ont été ajoutés. De plus, en accédant à la page "interactions" et en sélectionnant un bouton de type son ou vidéo, une interaction supplémentaire est disponible, permettant actuellement à l'utilisateur de mettre son son ou sa vidéo.

Modifier les informations:

Nom de l'élément :

Type de l'élément :

vous

☒ Bouton
L'élément créé sera un simple bouton

☐ Bouton son
L'élément créé sera un bouton qui lancera un bruit quand il sera utilisé

☐ Bouton vidéo
L'élément créé sera un bouton qui lancera une vidéo quand il sera utilisé

☐ Dossier
L'élément créé sera un dossier pouvant contenir d'autres boutons

Figure : Modification du type des boutons

Figure : Modifications des interactions des boutons

II.4 Corrections des bugs

Text Explication travail Bug

II.4.1 Erreur Print

Explication du Problème

Le bug du print est une erreur qui m'a été assigné lors de l'entretien que j'ai passé avec Jordan Arrigo. Dans l'option print des options il n'était pas possible d'avoir un autre texte en haut de la page et en bas de cette dernière et, le texte du haut de page pouvait être par dessus n'importe quel texte.

Analyse du Bug

J'ai donc dans un premier temps testé pour voir comment le bug se passait puis je suis allé voir où se situait la fonction

Méthode de Résolution

e

II.4.1 Phrase bug

Explication du Problème

e

Analyse du Bug

e

II.5 Optimisations de la synthèse vocale

II.6 Missions inaccomplis

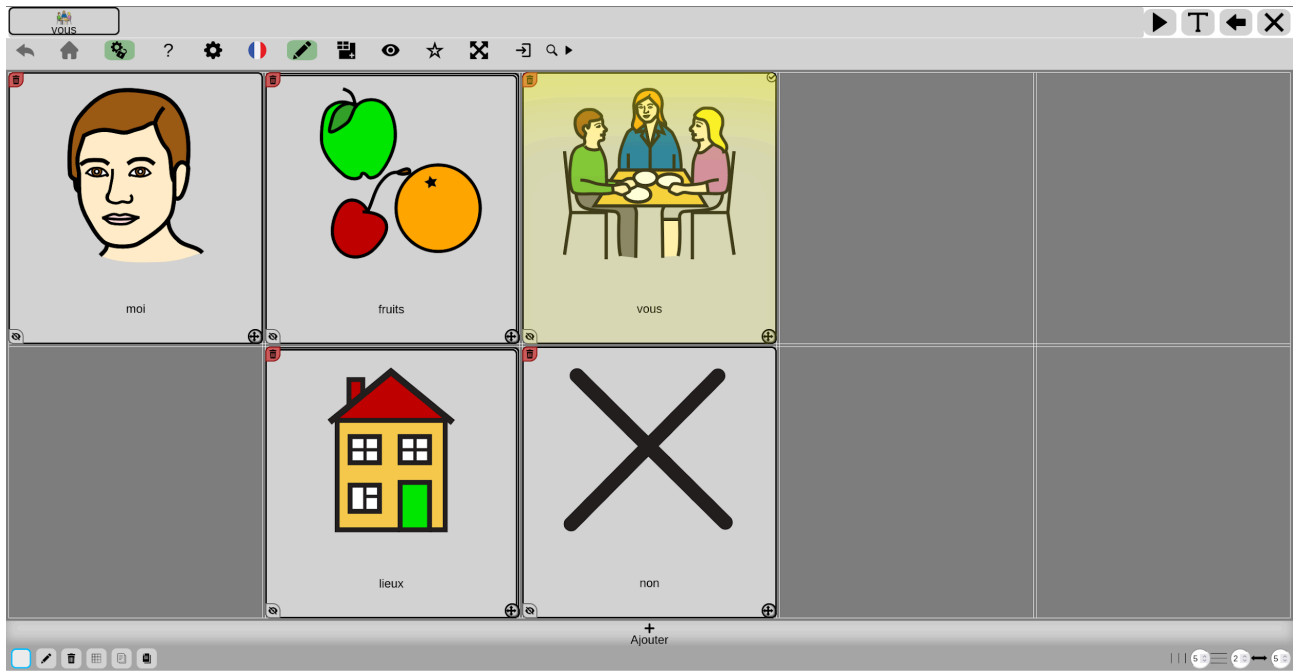
Ajout de vidéo/musique via api

Implémentation d'un système de conjugaison automatique

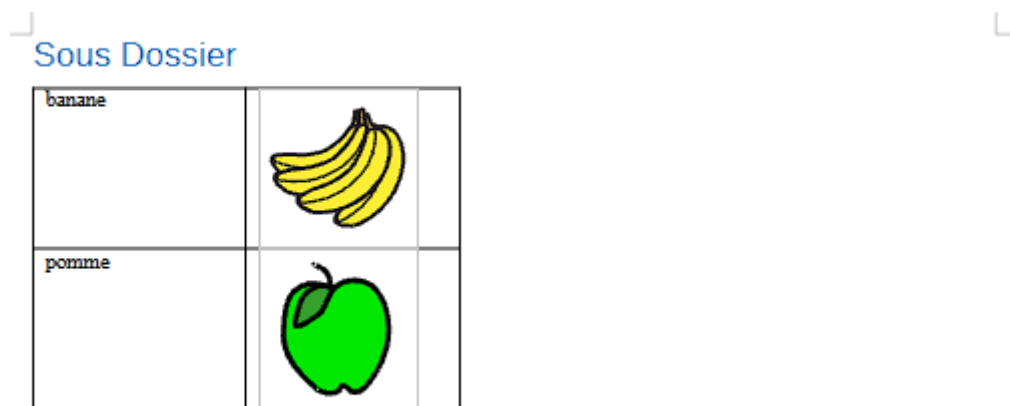
III. Conclusion

Glossaire

Annexes



Annexe : Mode Edition Page Principale



Annexe : Capture d'écran du rendu final sous dossier "fruits"

Résumé :

Mots clés :

Abstract :

Keywords :