

## **Treizième rencontre d'équipe - dernière démo - 31 juillet 2024**

*Tout le monde est présent*

### **Soutenance le 12 août à 9h**

*Commentaires/Questions clients finaux:*

- Inclure les limitations dans le rapport en plus du github
- Syntaxe style impression (1-; 1,2,3; 1-20; etc.) semblerait plus intuitif pour la modification de tempo
- Ç'aurait été bien d'avoir un retour (notification) qui dit au client pas de tempo détecté quand c'est le cas
- 

Planification finale / rapport / présentation / transmission du projet / user guide:

*TODO pour le client:*

- Backlog des issues encore existant
- Transmission du ownership du github
- Guide utilisateur
- Plan de projet et rapport pour identifier ce qui a été fait
- Procédure de signer la cession de projet sous license OpenSource

*TODO pour l'équipe:*

- Écrire le rapport (DATE LIMITE: **15 août 8h30**)
- Faire la présentation et le support
- Évaluation par les pairs (envoyer après la soutenance)

## **Douzième rencontre d'équipe - avant-dernière démo - 24 juillet 2024**

*Tout le monde est présent*

### **Soutenance le 12 août à 9h**

#### *Commentaires clients:*

- Tableau de tempos dans un fichier qui contient les tempos, permet d'en rajouter par la suite.
- Il faudrait peut-être une rétroaction plus claire du côté client lors de la progression (*nice-to/à voir*)
- Idéalement avoir un feedback qu'il n'y a pas de tempo trouvé et que c'est assigné au tempo par défaut
- Un checkbox/slider à l'outil de tempo UI pour overwrite un tempo s'il y en a déjà un présent/avoir un fall back.
- Avoir l'image docker pour la semaine prochaine.
- Peut-être documenter un peu la partie de modèle manuscrit pour une personne travaillant sur le projet dans le futur.
- ~~Un nice-to: avoir un % de correspondance entre le pdf et le fichier midi produit.~~

#### *Tâches réalisées:*

Jérémy: Travail sur les outils tempos côté UI

Laflèche: Travail sur les tests

Xavier: Évaluer les problèmes de bugs de parole en notes à la conversion (options Audiveris?).

Philippe: Ajout des outils tempo de Jeremy pour hardcode tempo avec le backend

Charlie: Travail sur la partie manuscrite et sur la sécurité

#### *Séparation des tâches pour la semaine du 25 juillet au 31 juillet:*

Tout le monde: QA sur le code

Jérémy: Dernier travail sur les outils UI - feedback sur le tempo non trouvé

Laflèche: Dockerfile/docker image

Xavier: QA plus intensif

Philippe: Option overwrite de tempo

Charlie: Job de performance 2 instances en même temps

## **Onzième rencontre d'équipe - 17 juillet 2024**

*Tout le monde est présent*

Bugs relevés à régler avant la démo:

- Paroles considérées comme des notes
- Avoir un bouton cancel
- Vérifier la validation de fichier / nom du fichier (sécurité)
- Nom de fichier pdf rendu en python, quelques bugs à vérifier

Autre nice-to avant la démo:

- Côté performance, combien de personnes veut-on pouvoir desservir (wide-release, juste les membres du club)
- Question sécurité - par rapport au code malicieux pouvant être intégré en pdf, sinon pour limiter les autres attaques, voir que le déploiement puisse être accédé par le VPN de l'école. S'assurer que les fichiers garbage donnés à l'application sont ignorés. (style fichier.exe.pdf) - (pdf a un *magic signature* avec pdf+version apparaît dans le premier kb du fichier) Pour le pdf - voir *Proof of concept or Get the Fuck Out vol. 7 - article 6*.
- Possibilité de sélectionner de retourner le fichier .mxl en plus du fichier midi.

*Tâches réalisées:*

Jérémy: Fichier Docker compose pour le front-end. Option pour forcer le tempo.

Laflèche: Test end-to-end en java.

Xavier: Conversion tempo XML fonctionnel

Philippe: Table de conversion en termes + aide Xavier

Charlie: Travail sur les partitions manuscrites - avec la reconnaissance de notes et symboles (pattern matching et entraînement ai). À travers quelques options, un certain niveau de reconnaissance.

*Séparation des tâches pour la semaine du 18 juillet au 24 juillet:*

Jérémy: Travail pour faire l'image docker.

Laflèche: Travail sur les tests du côté python.

Xavier: Travail avec Philippe.

Philippe: Ajout des outils tempo de Jeremy pour hardcode tempo avec le backend, évaluer les problèmes de bugs de parole en notes à la conversion (options Audiveris?).

Charlie:

## **Dixième rencontre d'équipe - 10 juillet 2024 - Rencontre client 5**

*Jérémy ne peut pas être là, Pascal non plus.*

Samuel Gagné est présent pour parler de l'hébergement du site web.

- Ce que les clubs utilisent habituellement Cédille, voir aussi avec le club AppIETS aussi. Si ça ne marche pas avec les clubs, re-communiquer avec Samuel Gagné pour l'hébergement.
- Les serveurs utilisés habituellement sont avec hébergement google, communiquer avec le STI pour les spécificités de ce qu'on a de besoin.
- Héberger l'application, les fichiers générés seront mis à part sur une BD du club ou sur l'ordinateur de l'utilisateur.

Nice-to à rajouter: Possibilité de sélectionner de retourner le fichier .mxl en plus du fichier midi.

### **Démonstration finale le 31 juillet.**

#### *Tâches réalisées:*

Jérémy: Outils front-end pour customiser la conversion dans le back-end, avec tempo forcé à partir du front-end.

Laflèche: Début de créations de tests JUnit.

Xavier: Chercher un moyen pour que Audiveris affecte le tempo correctement. Trouver une option d'export le fichier lu par Audiveris en XML pour le tempo.

Philippe: Travailler sur les problèmes de tempo, communication avec hbitteur, dev en charge de Audiveris. Début de travail sur l'alternative.

Charlie: Travail sur l'entraînement de partition manuscrite avec le training de Audiveris, permet seulement à détecter les symboles, mais pas les notes manuscrites.

#### *Séparation des tâches pour la semaine du 11 juillet au 17 juillet:*

Jérémy: Travail avec Xavier et Philippe.

Laflèche: Continuer à travailler sur les tests/la batterie de tests JUnit et possiblement travailler sur des tests Python pour la conversion.

Xavier: Sortir les tempos des XML audiveris et les envoyer au backend.

Philippe: Intégration de l'alternative XML pour le tempo avec Xavier.

Charlie: Va tenter de trouver une solution pour les partitions manuscrites pour la reconnaissance de notes.

### **Neuvième rencontre d'équipe - 3 juillet 2024**

*Tous les membres de l'équipe sont présents, les professeurs superviseurs sont les deux présents.*

*Hébergement du serveur pour le projet:* Camille propose sa machine comme hébergement, sinon on doit attendre le retour de Adrien pour savoir ce qui se passe de leur côté.  
Pour l'instant, on reste avec l'idée du déploiement par Docker.

Le travail d'implémentation et de développement est limité au plus tard le 2 août après la rencontre-démo 3 auprès d'Adrien.

Après le 1er-2 août, on fera le travail de présentation et de rapport final.

### **Soutenance le 12 août.**

*Tâches réalisées:*

Jérémy: Création d'un outil de sélection de pages voulues à être traduites (dans le pdf entrant).

Laflèche: Peu d'avancements de son côté pour les deux dernières semaines, un peu de travail d'évaluation de tests.

Xavier: *backburner* pour la dernière semaine, aide pour Philippe et Charlie.

Philippe: Ajouter la suppression des fichiers in et out une fois le fichier remis à l'utilisateur.  
Travail sur les tempos des partitions (*WIP*).

Charlie: Tests sur le modèle créé par Xavier pour les partitions manuscrites.

*Séparation des tâches pour la semaine du 4 juillet au 11 juillet:*

Jérémy: Continuer les outils front-end pour customiser la conversion dans le back-end.

Laflèche: Commencer les unit tests. Commencer les tests d'intégration (end-to-end tests).

Xavier: Travail sur le modèle de reconnaissance manuscrite.

Philippe: Continuer de travailler sur les problèmes de tempo, sinon ajouter des modifications possibles par l'utilisateur pour appliquer les tempos (Voir avec Jérémy pour faire un outil UI style  $[tempo = x \text{ à la mesure} = y]$  avec une liste possible de plusieurs tempos).

Charlie: Retourner dans Audiveris pour voir comment appliquer le modèle de reconnaissance manuscrite de Xavier et l'intégrer, voir si c'est faisable.

## **Huitième rencontre d'équipe 28 juin 2024 - Démonstration 1**

Tous les membres sont présents. Pascal, Adrien, Camille sont présents.

Présentation du premier prototype :

On a fait une démonstration live.

On a présenté les limitations et bugs d'Audiveris.

On a présenté les limitations de notre convertisseur de mxl à midi.

Pdf avec texte de chant -> cause des problèmes rendus en midi

Le client ne porte pas d'importance aux paroles/traduction.

Le client trouve important d'avoir le bon tempo. - > évaluer avec Xavier ses configurations, car il n'a pas le problème.

À terme le client aimerait une application (en apk privé peut-être) pourrait se prendre sur téléphone et enregistrerait sur le drive les partitions converties. Sinon le site web actuel semble être accepté.

Pour host le projet, Camille propose possiblement sa machine linux, sinon on va attendre du feedback de Adrien. Adapter notre documentation de déploiement selon ce qui sera décidé.

*Commentaire d'Adrien :* "Franchement excellent !" Pour ce qu'ils veulent faire, le prototype est suffisant. Curieux de voir la progression de la reconnaissance manuscrite. Bon boulot. Il espère que le problème de tempo va être réglé.

Un espace dans le nom du pdf peut causer des problèmes -> se fixe facilement.

*Séparation des tâches pour le sprint du 12 juin au 28 juin:*

Jérémy: Fix bug chrome, rajouter le champs tempo dans le FE et BE.

Laflèche: Finir d'évaluer les tests + créer des tests unitaires/end-to-end/d'intégration selon les balises évaluées.

Xavier: Travail sur la reconnaissance de partition manuscrite.

Philippe: Test de conversion de tempo avec language pack et options Audiveris (voir avec Laflèche)

Charlie: Travail sur la reconnaissance de partition manuscrite.

## **Septième rencontre d'équipe 19 juin 2024**

Tous les membres sont présents. Pascal est présent, Camille est indisponible.

### *Tâches réalisées:*

Jérémy: Fini de connecter avec le front-end.

Laflèche: Quelques tests. Plus de travail la semaine qui vient.

Xavier: Entraînement du modèle.

Philippe: Intégrer au back-end de Charlie avec son convertisseur Audiveris à mxl. et au front-end.

Charlie: Fini les bases du back-end.

### Remarque du prototype:

- Audiveris ne convertit pas le tempo automatiquement si par exemple le tempo est défini par un terme en allemand ou en italien. Donc définitivement soit réentraîner pour ça ou permettre aux utilisateurs de le modifier d'une façon quelconque
- Certain pdfs convertissent en fichier mxl qui sont invalides, donc rendu à la conversion midi, le script de conversion plante parce qu'il y a des erreurs, et le fichier mxl ne peut même pas être ouvert par musescore, donc identifier à quel point ça rive
- La façon que les fichier mxl sont lus versus les fichiers midi, il y a moins de nuance dans les fichiers midi, donc les crescendo/decrescendo et les différentes intensité ne sont pas lue pareil, je pensais que c'était mon script de conversion, mais je prends des fichiers de référence, ça fait la même chose
- En front-end, peut-être trouvé une façon de faire un progress bar, je sais pas trop, mais à voir, c'est vraiment pas une nécessité
- Il faudrait que l'équipe de tests vous testiez ce qu'on a fait, question de voir si vous relevez d'autres problèmes
- Certains fichiers mxl ne sont pas convertis (erreur Python)

### *Séparation des tâches pour le sprint du 12 juin au 28 juin:*

Jérémy: Connecter front-end au back-end avec Charlie.

Laflèche: Tests de partitions Audiveris.

Xavier: Training de partitions manuelles.

Philippe: Finir de connecter aux éléments du back-end. Tests d'intégration + tests unitaires.

Charlie: Finir les endpoints et rejoindre les éléments du front-end.

## **Sixième rencontre d'équipe 12 juin 2024 - Troisième rencontre client**

Tous les membres sont présents, Camille et Pascal présents, Adrien présent

### *Feedback sur le UI:*

- Simple mais clair

### *Tâches réalisées:*

Jérémy: Ajouter des fonctions de front-end, tester le back-end, feedback utilisateur.

Laflèche: Identification des éléments de Audiveris, documentation pour Audiveris pour préparation d'entraînement. Création de tests.

Xavier: Tester Audiveris et son code et d'autres solutions potentielles. Chercher des datasets de partitions écrites à la main. Commencer le training pour les partitions manuelles.

Philippe: Ajouter les fonctions de dynamiques pour le convertisseur mxl à midi. Intégrer au back-end de Charlie avec son convertisseur Audiveris à mxl.

Charlie: Création du back-end, convertisseur Audiveris à mxl, framework Gradle pour le projet en spring.

### *Séparation des tâches pour le sprint du 12 juin au 28 juin:*

Jérémy: Connecter front-end au back-end avec Charlie.

Laflèche: Tests de partitions Audiveris.

Xavier: Training de partitions manuelles.

Philippe: Finir de connecter aux éléments du back-end. Tests d'intégration + tests unitaires.

Charlie: Finir les endpoints et rejoindre les éléments du front-end.



### **Cinquième rencontre d'équipe 5 juin 2024**

Tous les membres de l'équipe sont présents.

#### *Tâches réalisées:*

Jérémy: Fait la base du front-end en react. Les fonctions placeholders sont là. Fonctionne sur pc et téléphone.

Laflèche: A travaillé sur faire des partitions de test pour les séries de tests avec Audiveris.

Xavier: Analyse et documentation pour ce qui est de Audiveris.

Philippe: Avancer un script pour faire une conversion de fichier MusicXML vers MIDI. Assez avancé pour les éléments de base nécessaires, peut être retravaillé pour ajouter des éléments/fonctions.

Charlie: A commencé l'intégration du back-end en java avec des endpoints.

#### *Séparation des tâches pour la semaine du 7 juin au 14 juin:*

Jérémy: Ajouter plus de feedback UI, plus d'éléments de contr

Laflèche: Finir les partitions de tests + des partitions à la main.

Xavier: Évaluer les set de training de Audiveris pour analyser les limites de ce qu'on peut faire.

Philippe: Finir le script MXL->MIDI et l'intégrer dans le back-end, aider sur le back-end.

Charlie: Finir les endpoints et connecter avec le système de conversion, et le front-end.

#### **Quatrième rencontre d'équipe/Deuxième rencontre client 29 mai 2024**

Tous les membres de l'équipe présents, Camille présente, Pascal présent vers 16h30.  
Adrien est présent pour la rencontre client.

Notes/Commentaires par rapport à la présentation initiale de Audiveris (par Xavier):

*Prendre en note que la présentation est faite sur l'ordinateur assez performant de Xavier.*

- Problème d'entraînement possible pour certaines erreurs -> proposition de Camille pour réentraîner et améliorer l'entraînement pour détecter les erreurs (il semble qu'en ce moment ce sont les notes fantômes/seule).
- Pour un minimum, Audiveris semble apprécié par le client.
- Le tempo doit probablement être retravaillé.
- Audiveris ne fait pas de fichier MIDI directement, donc à créer

Le projet doit fonctionner sur Windows (peut-être un nice-to d'être multi-plateforme).

Commentaires sur le rapport de projet:

- Bon français, bonne structure.
- 

#### *Séparation des tâches du sprint 1:*

Tous (ou Phil peu importe): Création du git/branch out à partir de Audiveris, + planning de tâches dans github.

Jérémy: Front-end de base React pouvant communiquer avec un API (architecture RESTful?)

Laflèche: Travail d'intégration et d'amélioration de Audiveris pour le projet

Xavier: Travail d'intégration et d'amélioration de Audiveris pour le projet

Philippe: Création du back-end avec Spring boot + export vers MIDI.

Charlie: Création du back-end avec Spring boot + pont Audiveris

#### *Séparation des tâches pour la semaine du 29 mai au 6 juin:*

Jérémy: Créer projet React, page d'accueil de base avec un module de téléversement de fichier pdf (et d'images compatibles) + commencer à faire des tests.

Laflèche: Identifier les éléments à garder, à améliorer, à épurer de Audiveris et commencer à apporter des améliorations selon l'ordre de priorité (à évaluer).

Xavier: Identifier les éléments à garder, à améliorer, à épurer de Audiveris et commencer à apporter des améliorations selon l'ordre de priorité (à évaluer).

Philippe: Créer les endpoints de communication avec front-end ou créer les endpoints/interface de communication et de contrôle avec Audiveris + Créer le module d'export pour Audiveris de leur type de fichier vers un format MIDI (car export vers MIDI non existant avec Audiveris - donc mxl ou omr file vers MIDI).

Charlie: Créer les endpoints de communication avec front-end ou créer les endpoints/interface de communication et de contrôle avec Audiveris

*Séparation des tâches pour la semaine du 7 juin au 14 juin:*

Jérémy: Ajouter des éléments UI/UX pour donner du feedback utilisateur. Tester l'intégration avec les endpoints API/communication front-end/back-end.

Laflèche: Continuer de travailler sur les améliorations à apporter.

Xavier: Continuer de travailler sur les améliorations à apporter.

Philippe: Finaliser les endpoints si nécessaire. Implémenter une gestion des erreurs pour les erreurs possibles à retourner au front-end (fichier invalide, erreur dans l'analyse, analyse incomplète, etc.) + tests unitaires et tests d'intégration.

Charlie: Aider sur l'optimisation de Audiveris et tester les améliorations apportées avec les partitions existantes.

### **Troisième rencontre d'équipe 22-05-2024**

Tout le monde est présent.

Ne pas oublier de faire des tests unitaires et d'intégration.

Veut-on un responsable des tests plus général? Chacun s'occupe des tests de ses parties et cette personne passe derrière.

Tester les technologies pour faire l'analyse des besoins et faire des revues de celles-ci.

Quelle technologie neuronale? CadenCV? Audiveris? Autre?

Quelle technologie front-end? React?

Proposition de Jérémie pour faire la revue par les pairs aux fins des sprints.

Faire les meetings de gestion de projet seulement aux 2 semaines/faire la partie gestion de projet là pour éviter le *overhead* de trop de meetings.

Les autres rencontres seront pour prioriser le développement du projet.

**TODO:** REMISE DU PLAN DE PROJET ET DOCUMENTS ASSOCIÉS AVANT OU AU MAXIMUM LE 29 MAI.

Trouver des partitions avec des particularités plus *chiantes* (armatures, clés, encadrement bizarre).

Pour la semaine prochaine, avoir une matrice de décisions avec poids décisionnels des différentes technologies à voir avec le client.

Quelle environnement OS (Windows? Linux? MacOS?)? (QUESTION CLIENT).

Comment évaluer la robustesse du produit obtenu? Calculer la fidélité selon quoi? (QUESTION CLIENT) -

Quoi faire valoir qu'il faut avoir à 100%, qu'est-ce qui peut être accepté à moins de 100% initialement? Quelles sont les *targets* absolus?

- Formulation d'un problème d'ingénierie (indicateur Q4-I1)
- Évaluation des enjeux économiques et sociaux (indicateur Q9-I1)
- Autres sections pertinentes à un plan de projet (indicateur Q11-I1)

??

- une description du contexte et de la problématique
- une description des objectifs du projet
- une description de la méthodologie qui sera utilisée au cours du projet
- une description de la composition de l'équipe ainsi que le rôle de chacun des membres
- une description du mandat incluant les livrables (résultats à fournir au client)
- une évaluation des enjeux économiques et sociaux
- une analyse des différentes solutions techniques pertinentes au projet
- une description de haut niveau des requis, contraintes, exclusions
- une planification sommaire selon la méthodologie
- un registre des risques
- une description des technologies qui seront utilisées pour la réalisation du projet
- une liste des références, si nécessaire

*Date prochaine rencontre d'équipe avec le client: **29 mai - 16h***

Tâches pour la semaine:

Faire une matrice décisionnelle avec poids et demander à Adrien s'il a des partitions existantes

- Philippe

Audiveris - Xavier

OpenCV/CadenCV - Jérémy

OMReader - Charlie

Oemer - Laflèche

### **Première rencontre client 15-05-2024**

Absence de Pascal - averti d'avance

Tous les membres de l'équipe sont présents.

Le client club de musique ÉTS, représenté par Adrien Misandeau, est présent.

Le projet est MAESTRO.

Le club fait des projets d'automatisation d'instruments qui jouent à partir de fichiers MIDI.

Projet-> faire une application qui peut recréer un fichier MIDI à partir d'une image de partition.

Donc faire une application qui permet de scanner les partitions, faire transition vers une image qui peut être transformée en fichier MIDI qui pourra être rejouer.

L'application se base sur une application qui utilise OpenCV. Mais il y a des problèmes avec la reconnaissance des lignes de la portée, particulièrement au-dessus et dessous. ne reconnaît pas:

Les doubles/triples bémols, dièses, les time signature.

-> donc manque de robustesse, soit nécessite beaucoup de tests des limites des fonctionnalités.

Pour ce qui est de la partition manuscrite -> Reconnaître du texte manuscrit sous forme de traduction.

Est-ce qu'analyse est real-time? **Non, le but est de générer le fichier MIDI à partir du scan et le donner à la base de données. La performance en temps (latence) n'est pas la priorité.**

Est-ce que la base de données est existante? **Non, donc on pourrait en créer une.**

**Proposition-> cloud. Peut être laissé en plug-and-play pour l'équipe du client suite au projet.**

Existence d'une base du projet? **Non, laissé à notre discrétion de comment faire l'algorithme, ou le prototype.**

**Le projet peut être testé à partir des fichiers MIDI sur application de lecture midi.**

Utilisateurs? **Dans un premier temps, les membres du club. Par après, à voir avec club si nécessaire.**

Quels sont les prototypes attendus par le client? **Pas particulièrement d'attente par rapport aux nombres de prototypes.**

**Il va y avoir au minimum deux démos auprès du client pour montrer les avancées du projet pour avec un feedback structurant qui permet de se réaligner si nécessaire.**

Demande du client: **Explorer fichiers MIDI qui peut avoir deux instruments qui pourraient être joués les deux en même temps. Aussi évaluer comment reconnaître deux instruments sur une même partition. Une fois que le projet est d'abord capable de faire un instrument sur un fichier MIDI.**

Prioriser un format de prototype plutôt qu'un format final qui peut être retravaillé par le client suite à la remise.

Utiliser Discord sous format Agile correct avec le client.

Exigences technologiques? **Python pour ce qui est de ce qu'ils font du côté du club étudiant.**

Documentation utilisateur? **Oui, mais dans un niveau où l'utilisateur connaît tout de même l'idée générale du projet.**

Préférence UI? **Plus fonctionnel que beau.**

Préférence du format de reconnaissance (PDF, autre)? **Du moment que la photo ou le scan marche bien, pas de préférence.**

Interface voulue? **Pas de préférence du client.**

*Proposition-> Back-end avec API, et le front-end peut être accessible web-base par ordinateur ou téléphone.*

Priorité de ce qu'on veut analyser? **Un maximum de fidélité à la partition, mais en ordre, au moins les bonnes notes au bon tempo, au bon moment, à la bonne force, même chose avec les pauses. Puis, les éléments comme crescendo/decrescendo, staccato, tenuto, etc.**

Commencer par les partitions de piano.

*Date prochaine rencontre avec le client: 29 mai - 16h*

*Date prochaine rencontre d'équipe: 22 mai - 16h*

**TODO:** Faire un tableau de tâches et Github et de distribution du travail.