

Projet PSAR :
Dispositif Autonome de Synthèse Sonore

Encadrant : Hugues Genevois
Cahier de Charges

Pierre Mahé & (Adrien Hug-Korda)

25 mars 2015

Table des matières

1	Contexte	2
2	Le produit	2
2.1	Description de la demande	2
2.2	Fonctions du produit	3
3	Contraintes	4
3.1	Contraintes materiel/logiciel	4
3.2	Contraintes fonctionnement	4
3.3	Contraintes d'extensibilité	5
3.4	Contraintes de Temps	5
4	Tests de validation	5
4.1	Test d'interaction	5
4.2	Test de l'interface d'utilisation	5
4.3	Test de duplication	6

1 Contexte

Maîtrise d'œuvre : Hugues GENEVOIS

Maîtrise d'ouvrage : Pierre MAHÉ, (Adrien HUG-KORDA)

Le projet DASS, Dispositif Autonome de Synthèse Sonore, est un nouveau projet de Michel RISSE, compositeur de Décor sonore ayant à son actif de nombreuses créations et dispositifs sonores. Ce projet est une collaboration avec Hugues GENEVOIS, Chercheur au LAM de l'Institut Jean le Rond d'Alembert.

Le projet consiste à mettre en place un dispositif permettant de le faire interagir avec son environnement dans le but de souligner les sons existant dans cet environnement. Le dispositif recueillera les informations nécessaire grâce à des capteurs, permettant de récupérer les bruits ambiants, la luminosité, le température...

Dans un premier temps, ce dispositif devrait être utilisé pour une installation artistique, en juillet prochain, dans le contexte de [nom de l'événement] à [nom du lieux]. Dans un second temps, Hugues GENEVOIS et Michel RISSE voudraient réaliser une installation plus importante de nombreux dispositifs avec pour chacun des comportements et des caractéristiques propre.

2 Le produit

2.1 Description de la demande

Le but du projet est de créer un dispositif électroacoustiques portable interagissant avec son environnement. Ce dispositif doit capter l'environnement qui l'entour à l'aide de capteurs diverses (microphone, luminosité, température...). Il devra traiter ces données pour en extraire des informations pertinent sur les changements se produisant autour de lui (chant d'oiseaux, passage d'une personne au alentour...). Grâce à ces informations, il pourra interagir en synthétisant des signaux acoustiques.

La synthèse devra être paramétrable, pour permettre au compositeur de modifier l'influence des informations récupéré des capteurs sur la synthèse.

Une interface utilisateur, permettant au compositeur de simuler des données captés, sera intégrer au dispositif. Cela lui permettra de faire des expérimentations pour sa composition.

Dans l'optique de la reproduction de ce dispositif pour des installations plus important. Une procédure devra permettre à un utilisateur de pouvoir reproduire l'ensemble du système, aussi bien niveau matériel que logiciel.

Le dispositif devant être portable, le programme doit fonctionner sur une nano-ordinateur de type Udoo Quad. De plus le langage du programme principal est PureData, pour assurer une maintenabilité et une extensionnalité plus facile.

2.2 Fonctions du produit

L'acquisition des données fourni par les capteurs devra être traité en temps réel pour permettre une bonne interaction avec son environnement. Les ressources du nano-ordinateur étant limitées, l'analyse des données devra être économe en temps de calcul, ainsi quand place occupée sur la carte mémoire.

La production de sons du dispositif devra être de la synthèse sonore et non une lecture de sons pré-enregistrés. De plus la synthèse sonore aura pour rôle de souligner les éléments déjà présent dans l'environnement et on ne faire un imitation de celle-ci.

Deux groupes de capteurs sont à distinguer, les capteurs qui vont simplement influencer sur les paramètres de la synthèse, comme le capteur de luminosité, de température...

Et le microphone qui lui à deux fonctions principal : extraire le motif rythmique et le motif mélodie.

Le motif rythmique devra être une sequence minimale, extrait d'un événement ponctuel ou d'une sequence rythmique période. Le motif mélodie, elle ne sera limité que par un nombre maximum de notes. Pour garder la structure de la mélodie, si il y en a une.

Les ressources du nano-ordinateur étant limité, la détection rythmique et mélodie ne pourra être qu'approximative. Pour la détection mélodie, le programme détectera uniquement la mélodie émergente (la plus forte) d'une mélodie polyphonique. Pour la détection rythmique, le programme ne pourra pas détecter un rythme périodique uniquement si le motif rythmique est exactement identique, sans impact parasite ou bruit. Une approximation rythmique sera tolérer tant qu'elle ne dépasse pas un certain seuil.

Liste des capteurs

- Microphone.
- Luminosité.
- Température.

- Pression.
- ???

La synthèse paramétrable devra être paramétrable. Ce paramétrage sera accessible par une interface graphique. Permettant à l'utilisateur de définir ce que chaque capteur contrôle comme paramètre de la synthèse. Pour plus d'ergonomie cette interface devra être exécutable à distance ou du moins sans avoir à brancher de périphérique (clavier ou écran) au nano-ordinateur.

Une seconde interface graphique devra permettre à l'utilisateur de fournir, à l'aide d'un fichier ou en entrant des valeurs dans un tableau, les données reçues de chaque capteur. Cela lui permettra de simuler le comportement du dispositif dans un environnement donné.

Certains paramètres variant très lentement comme la température ou la pression, l'utilisateur pourra également modifier la vitesse de lecture de ces valeurs.

La procédure détaillant toutes les actions permettant la reproduction du dispositif sera fournie pour la partie logiciel, ainsi que des schémas des divers montages électriques pour la partie matériel. En parallèle à ce tutoriel, une image de la carte mémoire du nano-ordinateur sera fournie. Ce qui permettra la duplication rapide pour le même type de carte (même modèle, même version).

Une fois le dispositif installé, il faudra pouvoir savoir, sans toucher au dispositif, ni brancher de périphérique, si il fonctionne correctement et si aucun capteur n'est défectueux.

3 Contraintes

3.1 Contraintes matériel/logiciel

Le programme doit fonctionner sur un nano-ordinateur Udoo et doit utiliser majoritairement les possibilités du langage PureData avec possibilité de réaliser des portions de code dans d'autres langages.

3.2 Contraintes fonctionnement

Le dispositif doit être autonome ; l'utilisateur allume le nano-ordinateur puis le système se charge et doit pouvoir s'exécuter pendant plusieurs heures (voir plusieurs jours) sans besoin d'une intervention quelconque.

Pour une future version, il serait envisagé que le dispositif soit autonome énergétiquement en embarquant par exemple des panneaux photo-voltaïques. Cela est doit déjà être prit en compte sur la première version.

3.3 Contraintes d’extensibilité

Le dispositif n’est qu’une première version qui devra être réutilisable pour des versions plus élaboré du projet, qui permettrait de faire communiquer les dispositifs entre eux pour les faire interagir ensemble.

Dans cet optique le projet doit être maintenable et extensible le plus facilement possible.

3.4 Contraintes de Temps

Le produit doit être livrée le 12/05/2015.

4 Tests de validation

4.1 Test d’interaction

Test

Ce test permet de savoir si le dispositif fonctionne bien.

Hypothèse

Le dispositif est déployé, tous les composants fonctionnent correctement.

Entrée

Néant.

Enchaînement nominal

1. L’utilisateur démarre le système. Il attend que le système soit prêt.
2. L’utilisateur fait une suite de plusieurs claquements de main.

Résultat attendu

Le système doit répondre à ces claquements de main en émettant un son en lien (rythme, fréquence) avec l’action de l’utilisateur.

4.2 Test de l'interface d'utilisation

Test

Envoi de données via l'interface d'utilisation pour simuler une réception particulière d'un capteur.

Hypothèse

Le dispositif est déployé, tous les composants fonctionnent correctement. Le nano-ordinateur est éteint.

Entrée

nom du fichier capteur_val.dat contenant un ensemble de valeurs.
Le capteur de température comme le capteur cible.

Enchaînement nominal

1. L'utilisateur démarre le système.
2. L'utilisateur envoie à l'aide de l'interface le fichier capteur_val.dat et choisi le capteur de température comme capteur correspondant ces valeurs.
3. L'utilisateur confirme.

Résultat attendu

Les valeurs ont été enregistrées par le dispositif. Le dispositif doit générer du son.

4.3 Test de duplication

Test

Ce test permet de voir si le tutoriel fourni avec le projet est suffisant pour reproduire le dispositif.

Hypothèse

L'utilisateur dispose de tous le matériel nécessaire pour faire le montage. De plus il possède un ordinateur sous Linux à sa disposition.

Entrée

Une carte microSD.

Enchaînement nominal

1. L'utilisateur fait les montages indiqués sur les schémas fournis avec le projet.
2. L'utilisateur suit la procédure d'installation détaillée dans le tutoriel.
3. L'utilisateur redémarre le nano-ordinateur.

Résultat attendu

Le dispositif fonctionne, les voyants des capteurs indiquent qu'ils fonctionnent.
Le dispositif émet des sons.