

GNG 1503

Génie de la conception

Manuel d'utilisateur du projet de conception

Le piano sans les maux

Travail présenté à

Pr. Emmanuel Bouendeu

Groupe FB-2: Civil+1

300106888 Muriel Maouad

300116859 Chris Mingele

300114995 Mario Moubarak

300116260 Elias Saab

8880758 Julie-Maude Lefrançois

Université d'Ottawa

5 avril 2020

Résumé

Dans le cadre du cours de GNG1503, Génie de la conception, on a été assignés la tâche de concevoir un système discret, économique et efficace pour mesurer et afficher la pression exercée par le pianiste sur chaque touche de piano ainsi que sa durée pour les chercheurs, professeurs de musiques et musiciens afin d'éviter les blessures aux pianistes et d'améliorer la pédagogie du piano. Suite à une rencontre avec notre client, Professeur à l'Université d'Ottawa, monsieur Gilles Comeau, le 21 janvier 2020, notre équipe a appliqué le processus de conception en génie afin de trouver une solution basée sur les besoins spécifiques du client. À l'aide de quelques itérations avec des prototypes, le prototype final a été mis au point.

Ce document traite du prototype en question de manière compréhensive afin d'établir de directives claires sur sa fabrication, son utilisation, la maintenance nécessaire ainsi que les étapes recommandées pour l'évolution future du projet.

Table des matières

Liste des figures	4
Liste des tableaux	5
1 Introduction	6
2 Construction du prototype	9
2.1 Catégorie	9
2.1.1 La boîte	9
2.1.2 Le mécanisme (circuit électrique, capteurs et USB)	10
2.2 La programmation d'une interface utilisateur	12
2.3 LDM (Liste des matériaux) et liste d'équipement (Tableau 1)	15
2.4 Instructions	15
2.4.1 Assemblage du mécanisme	15
2.4.2 Installation de la fenêtre de commande et démarrage des logiciels.	17
3 Comment utiliser le prototype	18
3.1 Navigation dans la fenêtre de commande (MATLAB App Designer®)	18
4 Comment maintenir le prototype	19
4.1 Facteurs météorologiques	19
4.2 Composantes fragiles	19
4.3 Source de puissance (par ordinateur)	20
5 Mesure de sécurité	20
6 Conclusions et recommandation pour les travaux futurs	20
7 Bibliographie et Appendice	21
7.1 Bibliographie	21
7.2 APPENDICES	21
Appendice I: Fichiers de conception	22

Liste des figures

Figure 1: Croquis du prototype final	7
Figure 2: Schéma de l'installation des capteurs	7
Figure 3: Branchement d'un premier capteur de pression	8
Figure 4: Aperçu de la fenêtre de commande proposée	8
Figure 5: Fichier Inkscape pour la boîte de bois	9
Figure 6: Captures d'écran montrant le code pour le branchement d'un capteur à Arduino	11
Figure 7: Exemple de code à notre disposition dans MATLAB	13

Liste des tableaux

Tableau 1 : LDM (liste des matériaux) et liste d'équipement

11

1 Introduction

Il a été observé, pour des années maintenant, que plusieurs pianistes souffrent de douleurs au dos, au cou, aux bras et/ou aux doigts causé par le jeu consécutif au piano. À vrai dire, les musiciens ont tendances à appliquer une pression sur une touche de piano qui est plus grande que celle nécessaire; la touche est aussi souvent maintenue pour une très longue période de temps, contrairement à ce qui est suggéré. Pour cette raison, les professeurs de musique cherchent à améliorer la pédagogie du piano en montrant aux joueurs la bonne technique dès le début de leur apprentissage.

Ainsi, notre prototype final s'agit d'un système à trois capteurs de pression, placé sous trois touches de piano distinctes qui calcule, sauvegarde et affiche à l'aide de graphiques les données de pressions récupérées lors du jeu de piano. Ce produit a été conçu en gardant en tête les détails important décrits par le professeur, monsieur Gilles Comeau, soit, la discrétion, la portabilité, la durabilité (fiabilité), le temps d'installation, la facilité d'utilisation, le coût, le poids et les dimensions. Le modèle est ciblé pour les chercheurs et chercheuses, les professeurs de musique ainsi que les musiciens. Nous croyons bien que notre conception aidera les chercheuses à faire comprendre aux joueurs que le fait d'appliquer, pour une durée plus courte, une pression qui est suffisante, mais qui n'est pas nécessairement grande, n'aura aucun effet sur le son émis OU Nous croyons que notre conception permettra aux chercheurs de corroborer ou non leurs hypothèse à savoir si les pianistes appliquent trop de pression sur les touches et, le cas échéant, à en ressortir des modèles courants et à prendre les mesures appropriées afin de prévenir la propagation de mauvaises techniques à l'avenir.

En prenant les critères initiaux en compte, ainsi qu'en appliquant le processus de conception en génie: la pensée conceptuelle, le produit final fut mis sur pied. Celui-ci combine des composantes physiques essentielles tel que des capteurs de pression, un arduino nano et un protoboard (liés à l'aide de câbles), ainsi qu'un affichage sur écran des données recueillies, et ce, tout en restant sous le budget de 100\$. Notre produit offre alors une interaction facile et compréhensible entre son utilisateur et les données de pressions.

Bref, le prototype construit par notre équipe (FB-02) s'agit du meilleur produit disponible dû à notre effort constant de répondre avec efficacité et efficience au besoin de notre client. En fait, la boîte qui contient notre modèle augmente la portabilité de ce dernier et protège le circuit davantage, les dimensions de celui-ci sont aussi significativement plus petites que celles des autres équipes grâce à l'utilisation d'un arduino de plus petite taille (type Nano), puis notre idée de créer selon les besoins du client une fenêtre de commande interactive offrant des options avancées en fait un produit de choix en comparaison aux systèmes qui ont été proposés par les

autres équipes. L'empathie et l'écoute que nous avons su démontrer vis-à-vis le problème du client jumelé au dévouement et à la créativité dont notre équipe a su faire preuve afin de viser à surpasser ses attentes ont fait de nous, un groupe prêt à surmonter avec enthousiasme tout défi qui s'est présenté lors du processus de conception.

Le prototype final :

Figure 1 : Croquis du prototype final

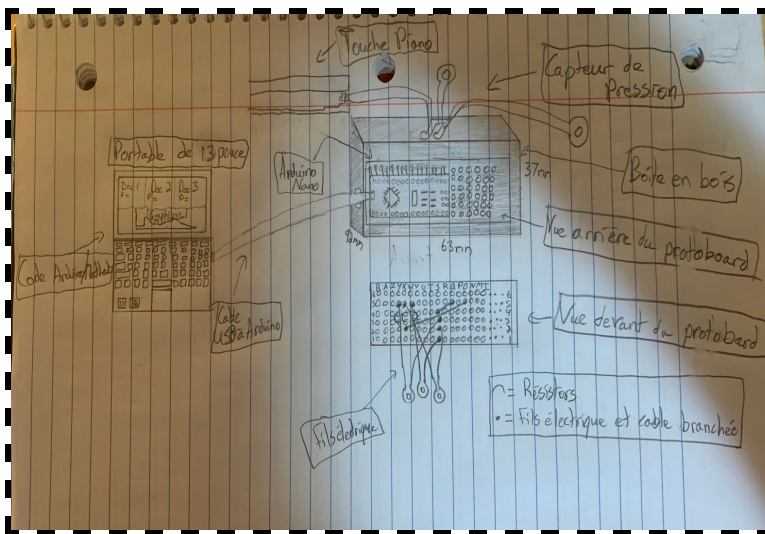


Figure 2 : Schéma de l'installation des capteurs

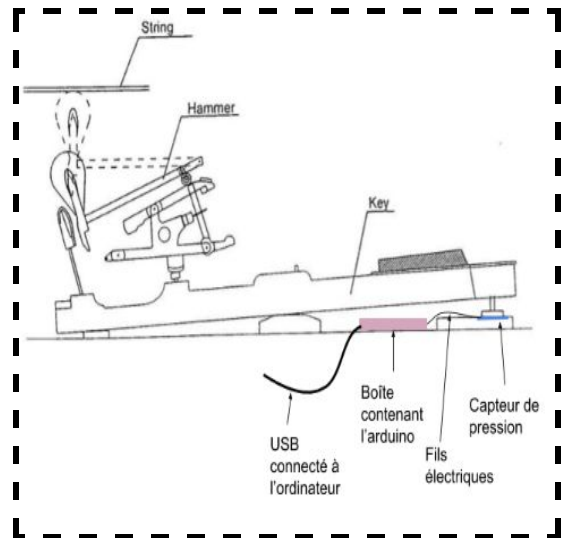


Figure 3 : Branchement d'un premier capteur de pression

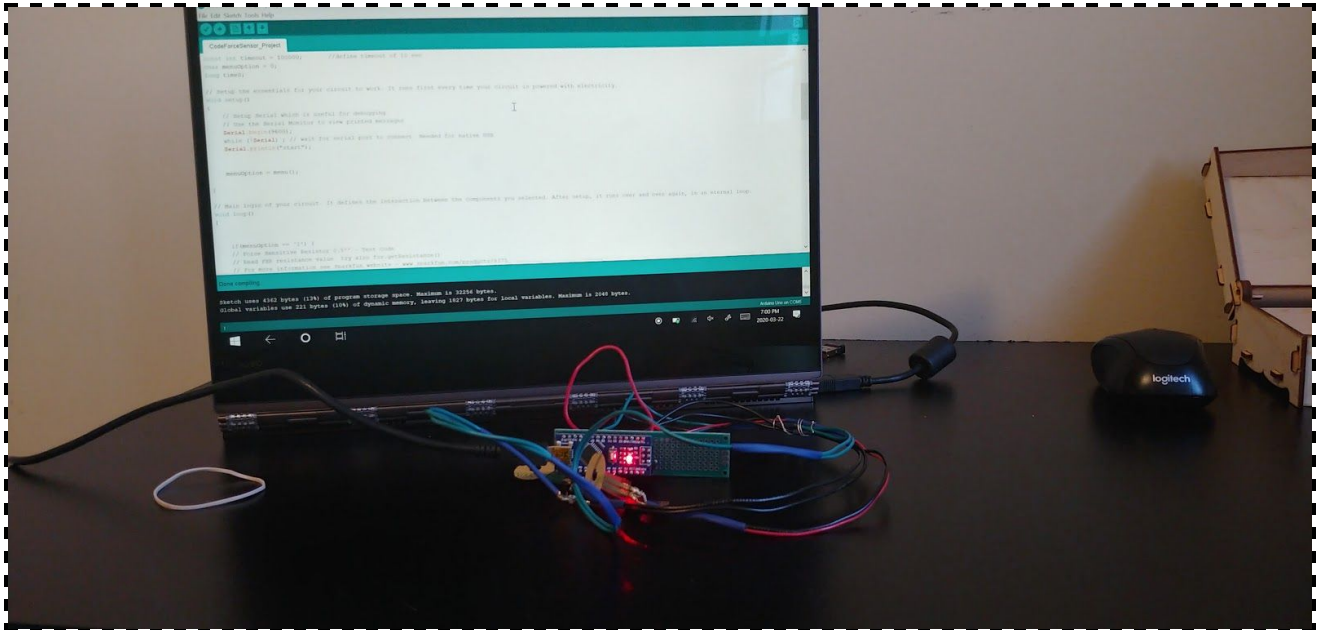
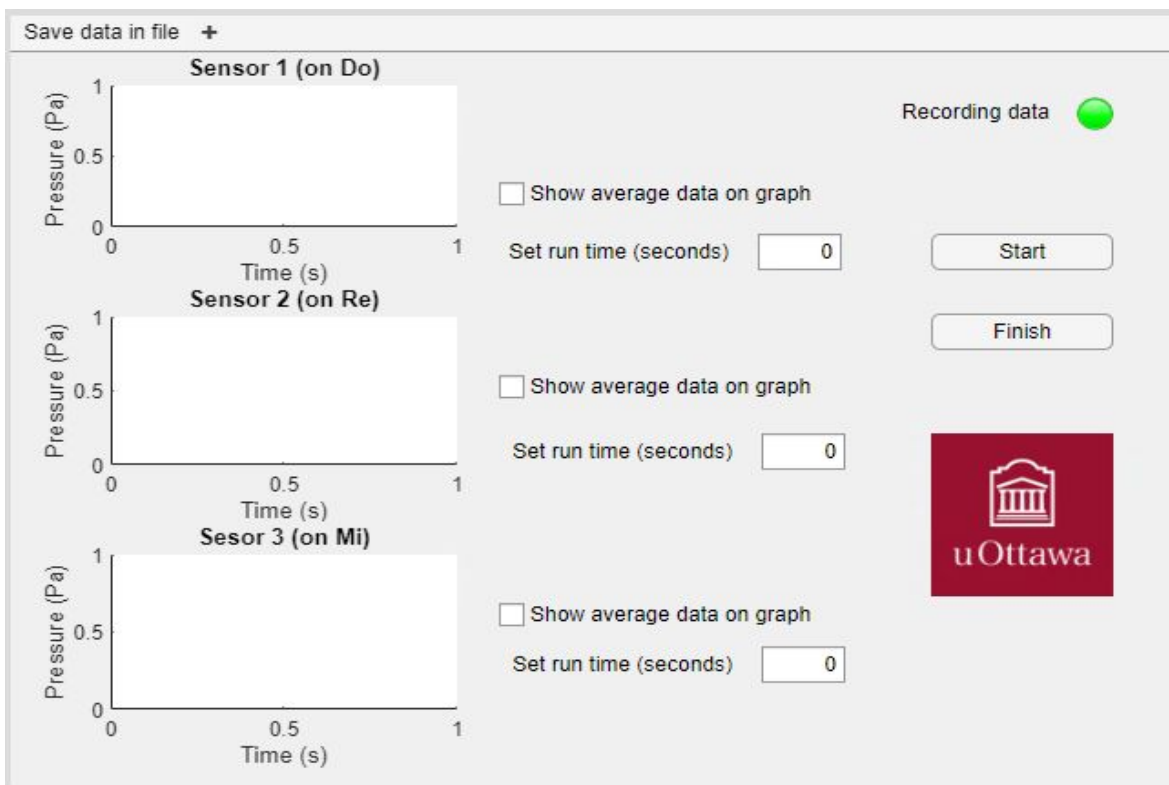


Figure 4 : Aperçu de la fenêtre de commande proposée



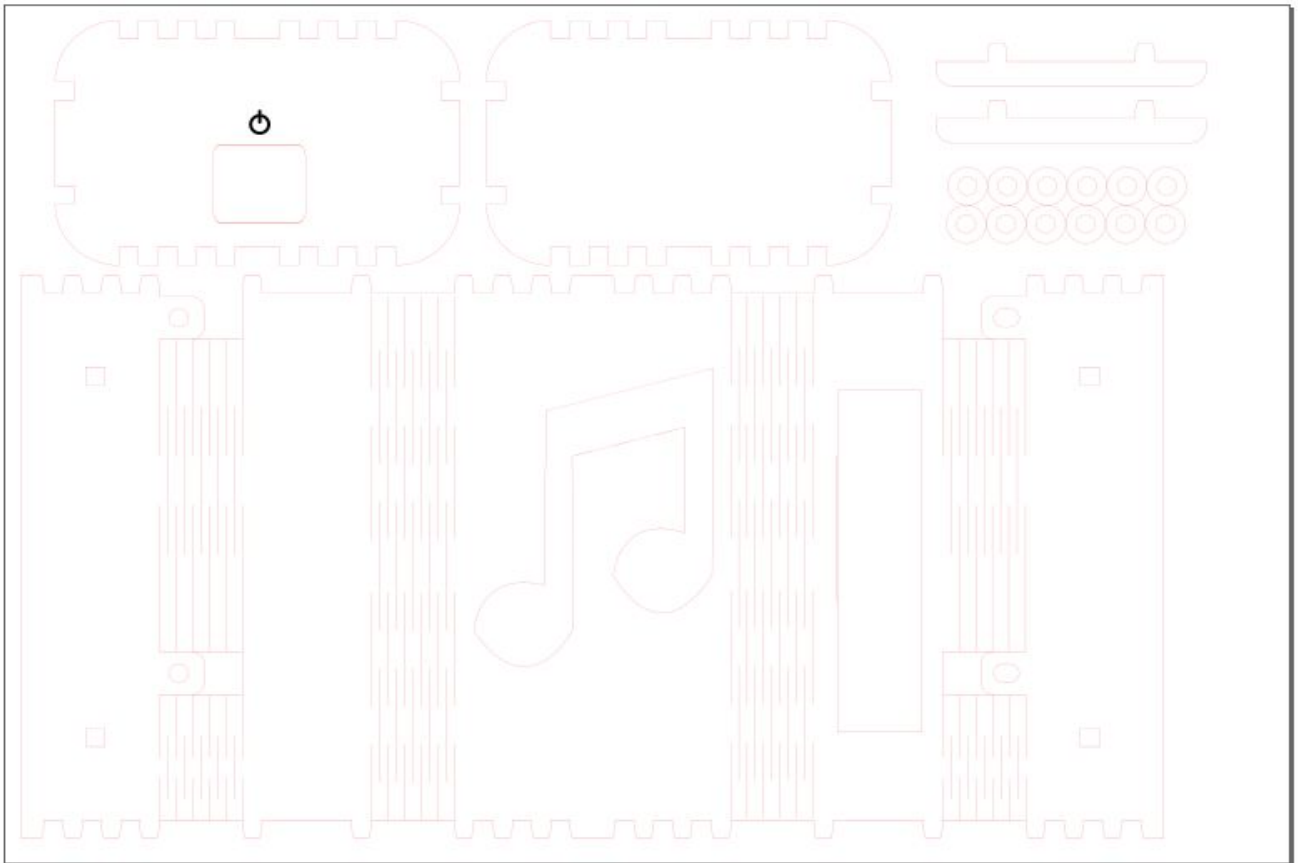
2 Construction du prototype

2.1 Catégorie

2.1.1 La boîte

Pour protéger les connections et la plaque d'arduino avec la manipulation du produit on devait concevoir un mécanisme pour couvrir complètement le produit. On a décidé de produire une boîte avec la machine de découpe laser qui va couvrir la plaque d'arduino et le protoboard. La boîte a été dessinée sur l'application Inkscape (compatible avec la découpe laser) avec les connaissances obtenues au cours des laboratoires. La boîte peut être modifiée pour répondre à toute contraintes physiques par rapport au milieu d'emplacement ou du matériel utilisé pour la construction de la boîte.

Figure 5: Fichier Inkscape pour la boîte de bois



Dimensions de la boîte de bois: 27mm de hauteur/63mm de longueur/92mm de largeur.

Nous avons conçu la boîte en mettant une image de note de musique sur le dessus pour qu'elle soit plus esthétique tout en représentant le projet. Elle a aussi été pensée pour que les fils des capteurs puissent sortir de celle-ci par des trous sur le côté. Ceci organisera les fils pour qu'ils ne traînent pas à l'intérieur de la boîte ou à l'intérieur du piano. Aussi, il y a un trou mis en place pour laisser le fils USB entrer dans la boîte pour alimenter le système au complet. Le trou est marqué avec un symbole « en marche » pour bien identifier son utilité dans le produit. Il y a aussi un trou sur un des côtés de la boîte pour avoir une visualisation de la plaque arduino et le protoboard. Ceci est aussi pratique pour vérifier le contenu de la boîte pour des mesures de sécurité et son entretien (ex: enlever la poussière, vérifier les composantes).

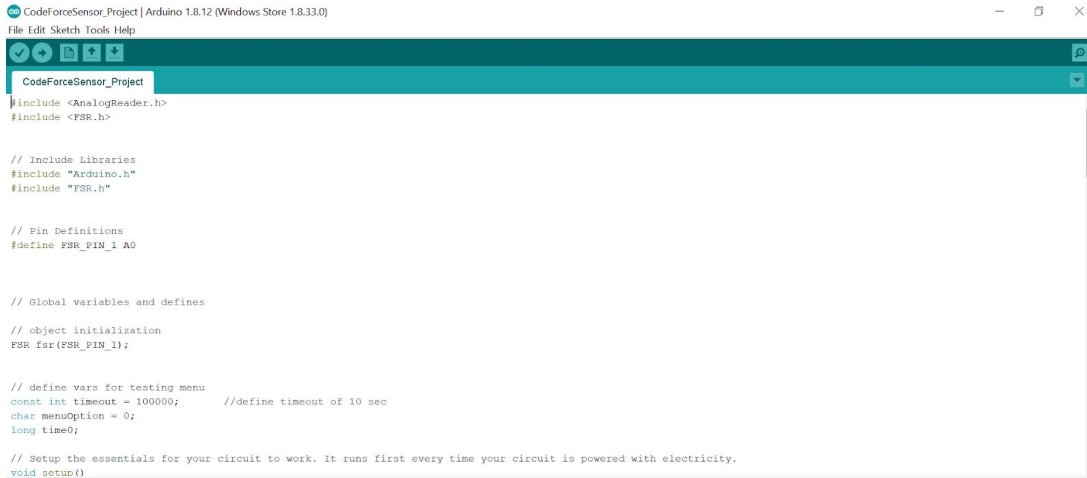
La boîte aurait été construite avec du bois coupé par la machine de découpe laser qui est coupée de façon à ce qu'on puisse plier le bois pour former la surface rectangulaire du prisme. Donc, au lieu d'avoir six pièces à manipuler pour construire la boîte, on en aurait eu juste trois, soit les deux bases et la surface. Ceci minimise le montant de composantes nécessaires pour construire la boîte et donc maximise la simplicité et la fiabilité du produit.

2.1.2 Le mécanisme (circuit électrique, capteurs et USB)

Le produit fonctionne en utilisant le programme introduit dans l'arduino pour lire les données créées par les capteurs de pressions qui sont connectés à l'arduino. Les trois capteurs de pressions sont alimentés par l'arduino lui-même alimenté en électricité par un fil USB qui est branché à l'ordinateur. Selon le circuit que nous avons eu le temps de tester en branchant notre premier capteur, nous avons calculé que trois résistances auraient dû être mises en place pour régler la puissance. L'ordinateur aussi joue le rôle d'interprétation des données puisqu'il présente les données sous forme de graphiques (un pour chaque capteur). Ceci aide à visualiser les tests de pressions sur les clés de piano pour simplifier le processus au complet.

Avec les capteurs alimentés à l'électricité, ils doivent être mis à terre et connectés à l'arduino encore une fois pour compléter le transport de l'information. L'information transportée est ensuite traitée par le programme transmis dans l'arduino et on reçoit les données qu'on peut comprendre en forme de graphique sur l'écran. Les capteurs de pression produisent l'information lorsque la pression est placée sur eux. Donc, lorsqu'ils sont placés sous les clefs de piano, les capteurs vont interpréter la pression exercée par le pianiste en chiffres numériques qui vont ensuite être mis dans un graphique pour visualiser la pression sur les clés désignées. La prochaine figure montre le code que nous avons utilisé pour brancher avec succès notre premier capteur de pression.

Figure 6 : Captures d'écran montrant le code pour le branchement d'un capteur à Arduino



```
CodeForceSensor_Project | Arduino 1.8.12 (Windows Store 1.8.33.0)
File Edit Sketch Tools Help

CodeForceSensor_Project
#include <AnalogReader.h>
#include <FSR.h>

// Include Libraries
#include "Arduino.h"
#include "FSR.h"

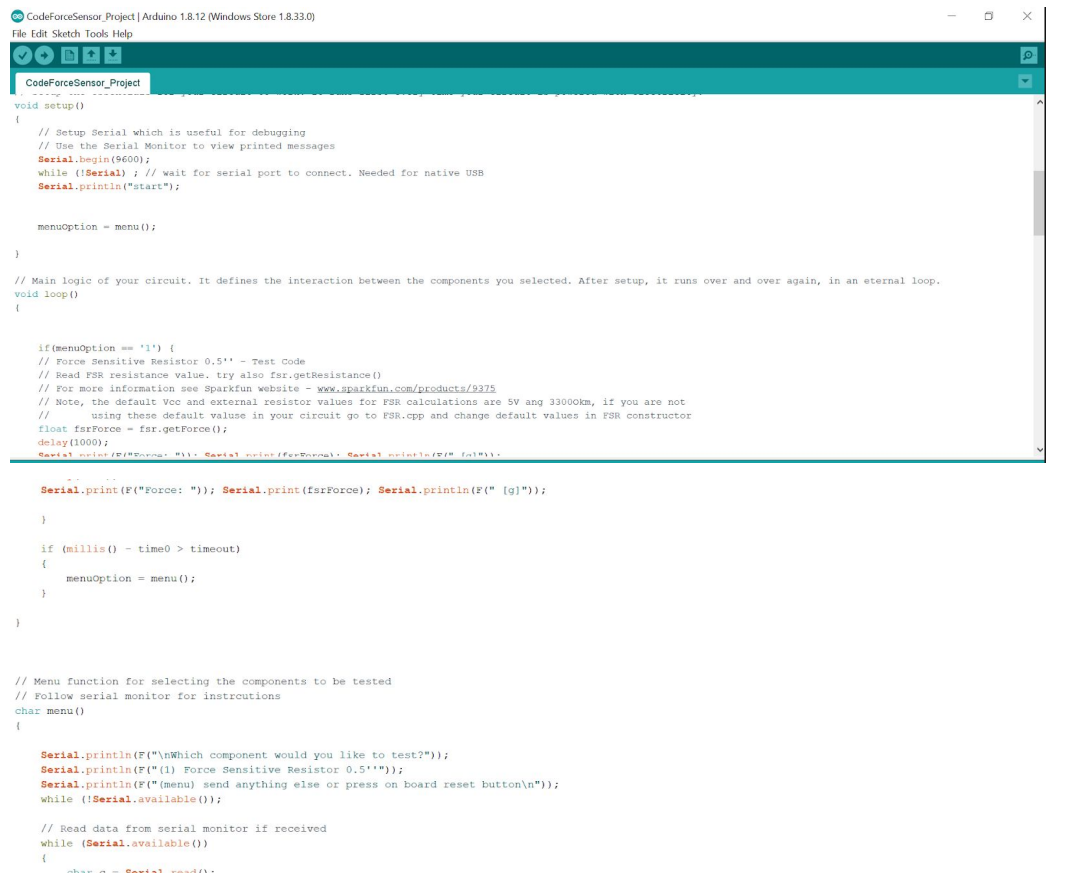
// Pin Definitions
#define FSR_PIN_1_A0

// Global variables and defines

// object initialization
FSR fsr(FSR_PIN_1);

// define vars for testing menu
const int timeout = 100000; //define timeout of 10 sec
char menuOption = 0;
long time0;

// Setup the essentials for your circuit to work. It runs first every time your circuit is powered with electricity.
void setup()
```



```
CodeForceSensor_Project | Arduino 1.8.12 (Windows Store 1.8.33.0)
File Edit Sketch Tools Help

CodeForceSensor_Project
void setup()
{
  // Setup Serial which is useful for debugging
  // Use the Serial Monitor to view printed messages
  Serial.begin(9600);
  while (!Serial); // wait for serial port to connect. Needed for native USB
  Serial.println("start");

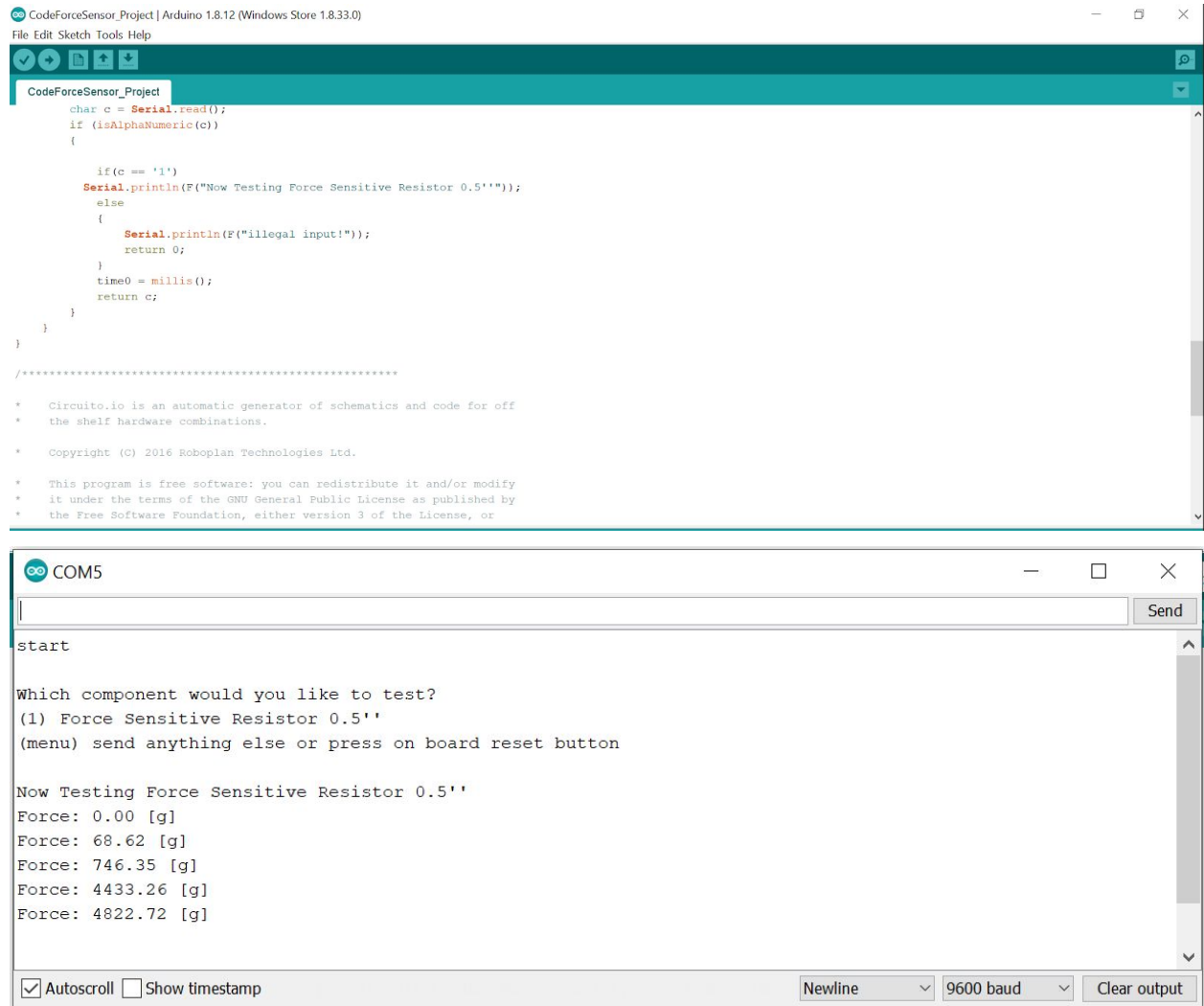
  menuoption = menu();
}

// Main logic of your circuit. It defines the interaction between the components you selected. After setup, it runs over and over again, in an eternal loop.
void loop()
{
  if(menuOption == '1') {
    // Force Sensitive Resistor 0.5" - Test Code
    // Read FSR resistance value, try also fsr.getResistance()
    // For more information see sparkfun website - www.sparkfun.com/products/2225
    // Note, the default Vcc and external resistor values for FSR calculations are 5V and 33000Ohm, if you are not
    // using these default value in your circuit go to FSR.cpp and change default values in FSR constructor
    float farForce = fsr.getForce();
    delay(1000);
    Serial.print(F("Force: ")); Serial.print(fsr.getForce()); Serial.println(F(" [g]"));
  }

  if (millis() - time0 > timeout)
  {
    menuoption = menu();
  }
}

// Menu function for selecting the components to be tested
// Follow serial monitor for instructions
char menu()
{
  Serial.println(F("\nWhich component would you like to test?"));
  Serial.println(F("(1) Force Sensitive Resistor 0.5\""));
  Serial.println(F("menu) send anything else or press on board reset button\n"));
  while (!Serial.available());

  // Read data from serial monitor if received
  while (Serial.available())
  {
    char c = Serial.read();
```

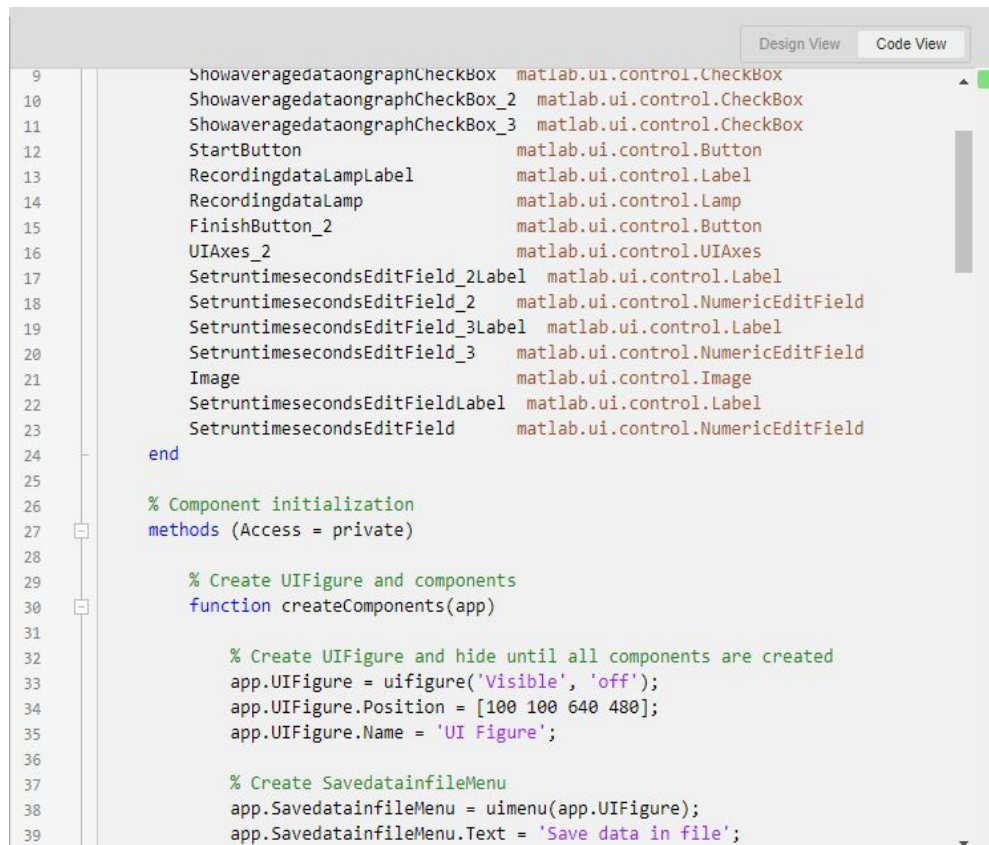


2.2 La programmation d'une interface utilisateur

Comme élaboré dans les livrables précédents, notre équipe a choisi de créer une fenêtre de commande interactive (de type GUI) à l'aide de la fonction AppDesigner dans MATLAB. Il est important de mentionner qu'au moment de la fermeture de l'université due au COVID-19, nous étions rendus à faire installer la licence pour connecter arduino à MATLAB. C'est donc la raison pour laquelle il nous manque des parties du code ou qu'il est tout simplement inexistant pour les fonctions avancées. L'éclatement de notre équipe et l'impossibilité d'obtenir la collaboration de nos pairs à distance pour le genre de travail que nous voulions faire nous a forcé à mettre de côté la tâche de la programmation. Toutefois, nous avons trouvé sur le site de

MathWorks (la compagnie produisant le logiciel MATLAB), un code¹ très semblable qui présente sous forme de graphique en temps réel les données de température récoltées par un capteur. Il aurait simplement fallu modifier un peu les paramètres du code pour convenir à notre objectif, soit récolter et interpréter en temps réel sous forme de graphiques les données émises par trois capteurs de pression placés sous les touches. L'avantages d'utiliser l'App Designer, et ce qui nous distingue des autres équipes à ce point, est que les différentes options avancées comme la création de graphiques et de différents boutons de commande et effets spéciaux offerts dans la librairie des composantes ont leur code déjà intégrés dans le logiciel (voir exemple figure 7). Il ne suffit que de les modifier pour répondre à nos besoins. Le temps que nous aurions sauvé pour la programmation grâce à App Designer nous aurait permis de passer plus de temps sur la programmation d'options facultatives et plus avancées. Ainsi, en offrant au client le choix entre quelques options, et en discutant avec le client de ses préférences par courriel, nous aurions pu personnaliser la fenêtre de commande selon ses désirs et ainsi surpasser les attentes de base de celui-ci.

Figure 7: Exemple de code à notre disposition dans MATLAB



```

9      ShowaveragedataongraphCheckBox matlab.ui.control.CheckBox
10     ShowaveragedataongraphCheckBox_2 matlab.ui.control.CheckBox
11     ShowaveragedataongraphCheckBox_3 matlab.ui.control.CheckBox
12     StartButton matlab.ui.control.Button
13     RecordingdataLampLabel matlab.ui.control.Label
14     RecordingdataLamp matlab.ui.control.Lamp
15     FinishButton_2 matlab.ui.control.Button
16     UIAxes_2 matlab.ui.control.UIAxes
17     SetruntimesecondsEditField_2Label matlab.ui.control.Label
18     SetruntimesecondsEditField_2 matlab.ui.control.NumericEditField
19     SetruntimesecondsEditField_3Label matlab.ui.control.Label
20     SetruntimesecondsEditField_3 matlab.ui.control.NumericEditField
21     Image matlab.ui.control.Image
22     SetruntimesecondsEditFieldLabel matlab.ui.control.Label
23     SetruntimesecondsEditField matlab.ui.control.NumericEditField
24
25     end
26
27     % Component initialization
28     methods (Access = private)
29
30     % Create UIFigure and components
31     function createComponents(app)
32
33     % Create UIFigure and hide until all components are created
34     app.UIFigure = uifigure('Visible', 'off');
35     app.UIFigure.Position = [100 100 640 480];
36     app.UIFigure.Name = 'UI Figure';
37
38     % Create SavedatainfileMenu
39     app.SavedatainfileMenu = uimenu(app.UIFigure);
40     app.SavedatainfileMenu.Text = 'Save data in file';

```

¹ GOVINDARAJAN, Madhu. «Plotting Live Data from a Temperature Sensor Using Arduino and MATLAB».

Un autre avantage que notre fenêtre de commande créée dans App Designer aurait eu sur nos concurrents est que la nôtre aurait été interactive, donc l'utilisateur aurait pu la contrôler. De notre côté, l'équipe aurait fait tout en son pouvoir afin que ce contrôle par l'utilisateur soit le plus intuitif possible, avec des options de commande claires, pour ne pas que cet atout ne tourne en un désavantage. La figure 4 représente bien une des options que nous allions proposer ainsi que des options avancées possibles parmi lesquelles on y trouve la possibilité de sauvegarder les données dans un fichier, de définir un temps d'exécution pour la récolte des données, de montrer la valeur moyenne de la pression sur l'intervalle de temps, de commencer et terminer la récolte de données, l'affiche d'un voyant lumineux qui indique la récolte des données, etc.

Il est à noter que le texte aurait pu être en français ou en anglais ou bilingue selon ce que le client nous aurait communiqué par courriel. Également, comme il est possible de constater dans la figure 4, le code aurait prévu de convertir les données brutes récoltées en grammes par centimètres carrés afin de les présenter au client en Pascals. Cette étape représente un simple calcul arithmétique et aurait été simple à programmer ($1 \text{ g/cm}^2 = 0.010197162129779$).

Au départ, nous souhaitions explorer l'option de transmission des données par Bluetooth. Nous avons jamais mentionné pourquoi nous avons jugé que cela était une mauvaise idée. Celle-ci semblait bonne au départ, car elle aurait certainement permis un meilleur esthétisme et peut-être un moins gros volume à placer dans le piano au final. Toutefois, nous avons suivi les conseils des experts, dont notre TA David, qui nous l'a déconseillé. Puis, nous avons pesé les pour et les contre et nous avons remarqué que d'installer un système bluetooth aurait certainement fait augmenter le prix total de notre prototype et que les défaillances étaient plus probables et moins faciles à régler si l'on transmettait les données de cette manière, ce qui n'est pas avantageux pour le client au final, si on a pu faire la même chose en répondant aux contraintes imposées par le problème haut la main sans cette fonction. Notre option concernant notre choix de transmission des données par USB semble donc plus avantageuse.

Ce que nous avons réussi à faire au final pour le troisième prototype est de récolter les données d'un seul capteur de pression. Alors, le même modèle aurait pu être recréé pour les deux autres. Pour cette partie, il ne restait qu'à terminer la programmation, c'est à dire remplir les trous dans le code afin d'interpréter les données récoltées par nos capteurs de façon graphique grâce à App Designer, communiquer avec le client et ajouter les différentes options. Nous sommes confiants que nous aurions pu terminer et peaufiner les derniers détails dans les délais demandés, si ce n'était pas de la situation particulière dans laquelle nous nous trouvons.

2.3 LDM (liste des matériaux) et liste d'équipement

Tableau 1 : LDM (liste des matériaux) et liste d'équipement

Produit/matériaux	Quantité/Unités	Équipements
Arduino nano (via Amazon)	1	Logiciel Arduino
Protoboard (via Makerlab)	1	Logiciel MatLab
Capteur de pression FSR404 (via Digikeys)	3	Équipement de soudure
Résistance (N/A)	3	
Paquet de fils électriques (via Makerlab)	1	
Cable USB à Arduino (via Elie)	1	
Boite en bois (N/A)	1	
Ordinateur portable (via Mario)	1	
Heat Shrink (via Makerlab)	9	
Cable Homme-Homme et Homme-Femme (via Makerlab)	10	

2.4 Instructions

2.4.1 Assemblage du mécanisme

Pour assembler le mécanisme, on doit s'assurer qu'on possède tout les composantes nécessaires pour le fonctionnement du produit. Étant donné que la plupart des composantes sont déjà connectées ensemble sur le protoboard, il n'est pas très difficile d'assembler le reste du mécanisme. Une première partie consiste à s'assurer que le câble USB est bien connecté de

l'ordinateur à l'Arduino, car celui-ci fournit le circuit en électricité. L'Arduino est ensuite branché dans le protoboard à l'aide des connecteurs femelles à mâles qui sont soudés sur le protoboard. Le protoboard contient déjà toutes les connections d'étain nécessaires pour alimenter les capteurs de pression avec la puissance de l'arduino ainsi que pour transporter l'information des capteurs à l'ordinateur portable. On doit juste s'assurer que le tout est en bonne forme et qu'il n'y a pas de problèmes avec les connections et les fils dans le produit.

Les étapes d'installation détaillées du mécanisme sont les suivantes :

1. Soulever les touches et glisser chacun des trois capteurs sous une touche à travers la tige de métal en s'assurant de la placer sous le coussin amortisseur.
2. Fermer les touches.
3. Placer les fils reliant les capteurs à l'arduino de façon à ce qu'ils sortent des touches de façon ordonnée et droite vers l'intérieur du piano sans encombrer les autres touches.
4. Ajuster l'égalité des touches.
5. Placer l'arduino dans la boîte de bois.
6. Fixer la boîte de bois contenant l'arduino avec du ruban adhésif à l'intérieur de la caisse de résonance.
7. Brancher les fils des capteurs à l'arduino et ranger l'excès de fil dans la boîte de bois.
8. Brancher le fil USB de l'arduino à l'ordinateur.
9. Sécuriser le fil USB à la surface sur laquelle il se trouve à l'aide de ruban adhésif.

Il est important de noter que pour déplacer le mécanisme, il faut d'abord le désinstaller en effectuant les étapes suivantes avant de procéder à nouveau aux étapes précédentes:

1. Débrancher le câble USB de l'ordinateur et de l'arduino.
2. Enlever soigneusement le ruban adhésif tenant le câble USB à l'intérieur du piano.
3. Débrancher avec caution l'extrémité des capteurs de pression reliée au circuit dans la boîte en bois.
4. Enlever soigneusement le ruban adhésif pour déplacer la boîte en bois contenant l'Arduino.
5. Soulever les touches et enlever les capteurs de pression de sous le coussin en les manipulant de façon très délicate. S'assurer de remettre le coussin à sa place.
6. Fermer les touches.

Précisions sur la manipulation et l'installation du mécanisme :

La manipulation du système (on parle ici du « hardware ») avec sécurité est très simple et se fait à deux niveaux. D'abord, il faut s'assurer que les connections sur le protoboard et les capteurs de pressions soient encore présentes et non détachés. Ceci assure le fonctionnement du produit et la circulation de l'électricité avec sécurité dans le produit. Avec le protoboard sécurisé, on peut alors brancher l'Arduino dedans.

De plus, c'est seulement lorsque l'arduino est branché dans le protoboard avec sécurité (soudage terminé) qu'on peut les installer dans la boîte. La boîte sert à organiser les fils donc il faut s'assurer que les fils sortent de l'image au dessus de la boîte et qu'ils ne sont pas empêtrés ensemble. Avec les fils organisés et la boîte fermée, on peut installer le tout dans le piano en s'assurant que les capteurs rentrent dans les barres métalliques et que les connections des fils aux capteurs ne touchent pas le feutre sous les clés.

2.4.2 Installation de la fenêtre de commande et démarrage des logiciels.

Voici les étapes à suivre afin de démarrer le logiciel et d'afficher la fenêtre de commande interactive qui servira à recueillir les données :

1. Une fois le câble USB branché à l'ordinateur, il faut allumer ce dernier.
2. Accéder au bureau en se connectant avec un compte usager.
3. Si vous utilisez un ordinateur connecté au réseau UOttawa, ouvrir MATLAB dans la barre de recherche Windows.

OU

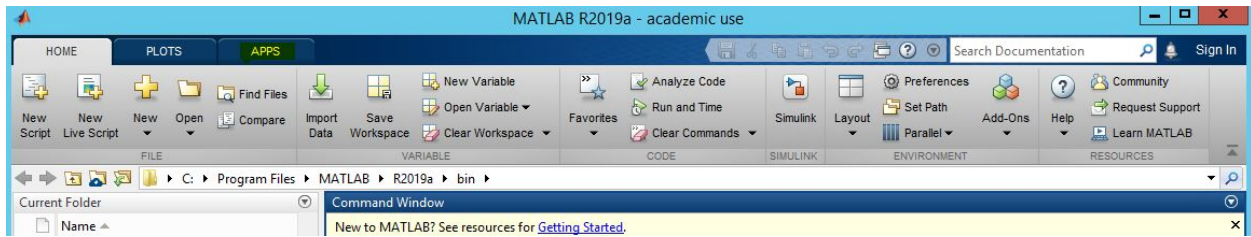
Si vous utilisez un ordinateur portable ou personnel, ouvrir une page Internet, Google Chrome ou Firefox et accéder au site de Remote Apps de l'Université d'Ottawa :

<https://remoteapps.genie.uottawa.ca/RDWeb/Pages/en-US/login.aspx?ReturnUrl=/RDWeb/Pages/en-US/Default.aspx>. Entrer son identifiant et mot de passe personnel afin d'accéder au réseau UOttawa (l'identifiant précédant @uottawa.ca et le mot de passe associé)².

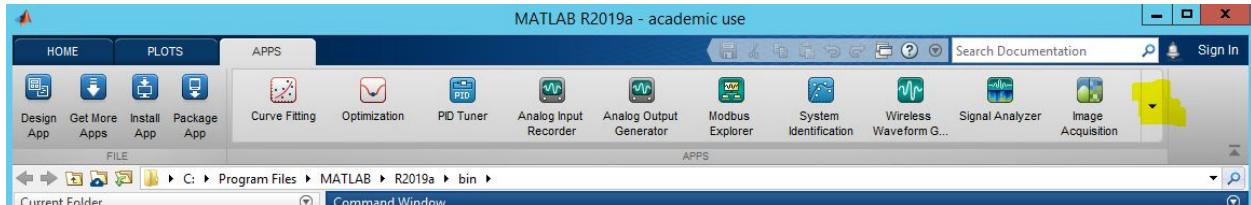
4. Ouvrir MATLAB. Si l'application ne veut pas ouvrir ou envoyer un message d'erreur, les étapes à suivre sont les suivantes :
 - 1) Déconnecter votre compte de Remote Apps et reconnecter de nouveau.
 - 2) Redémarrer l'ordinateur.
 - 3) Si le problème persiste, contacter le service des *Technologies de l'information* de l'université. Vous pouvez appeler au numéro sans frais, faire une demande en ligne ou encore vous rendre au bureau de service au rez-de-chaussée du bâtiment Morisset (MRT) en personne.

² Le réseau uOttawa est hautement sécurisé.

5. Dans la fenêtre de commande en haut à gauche, cliquez sur APPS.



6. Cliquer sur la flèche du menu déroulant «Show more».



7. Taper le nom de l'application de récolte de données dans la barre de recherche (Piano_sans_les_maux).

8. Ouvrir l'application en cliquant dessus. Lorsque l'application est exécutée à partir de l'invite de commande, il faut s'assurer que le fichier est dans le dossier en cours ou dans le système de fichiers MATLAB.

3 Comment utiliser le prototype

3.1 Navigation dans la fenêtre de commande (MATLAB App Designer ®)

En guise de rappel, la figure 4 est un exemple d'une fenêtre de commande que nous aurions proposé au client si le cours du projet s'était déroulé normalement. Une fois l'application Piano_sans_les_maux installée et ouverte, la navigation dans la fenêtre de commande (lancée par l'application) devrait être assez simple. Notre équipe a travaillé fort à imaginer pour le client une expérience interactive positive et intuitive en ce qui a trait à l'utilisation de la fenêtre de commande pour la récolte des données. Toutefois, voici quelques lignes directrices à suivre (supposément que le produit aurait été terminé) :

Démarrage/arrêt de la récolte de données par les capteurs :

- Pour commencer la récolte de données, cliquer sur le bouton « Start ». Si tout fonctionne bien un voyant lumineux vert s'allume en haut à droite de la fenêtre de commande et indique la récolte de données en cours («Recording data ») et des données apparaissent sur les trois graphiques.

- Pour arrêter la récolte de données, cliquer sur le bouton « Finish ». Les données devraient alors cesser de se générer sur les graphiques. N'oubliez pas d'enregistrer les données dans un fichier avant de démarrer une nouvelle exécution.

Sauvegarde des données dans un fichier :

- Pour sauvegarder les données dans un fichier, cliquer sur le menu « Save data in file » en haut à droite de la fenêtre de commande. Une nouvelle fenêtre de commande devrait apparaître. La prochaine étape est de déterminer l'emplacement de votre fichier, son nom et son format (.txt ou .xls).³ Cliquer sur « Save all ».

N.B.: Le fichier contenant les données devrait se trouver à l'emplacement choisi à l'étape précédente.

Options avancées :

- Pour définir une limite sur le temps d'exécution de la récolte des données, cliquer dans la boîte « Set run time (seconds) » et changer le 0 pour le temps désiré en seconde.
- Pour afficher la donnée moyenne sur le graphique, cocher la boîte « Show average data on graph ».

4 Comment maintenir le prototype

4.1 Facteurs météorologiques

Le mécanisme n'a pas été conçu pour résister à certains phénomènes météorologiques tels que les intempéries. En d'autres mots, la pluie, la neige, et même le froid ainsi que l'exposition à un fort ensoleillement sont tous des facteurs qui posent un grand risque sur la fonctionnalité de l'appareil. Cela étant dit, le dispositif devrait être utilisé à l'intérieur à une température de la pièce et à l'écart de la nourriture et des substances liquides.

4.2 Composantes fragiles

En raison du fait que certains des composants du dispositif sont très fragiles telle que l'arduino nano, il a fallu qu'on trouve un moyen d'isoler l'appareil dans un conteneur afin de le protéger de tout dommage. Ainsi, nous avons conçu une boîte pour abriter l'appareil et lui permettre

³ Note au correcteur : Le fichier aurait pu être enregistré sous le format .txt et, si cela avait été possible, en .xls (compatible avec Excel). Nous devons être honnêtes et dire que nous n'avons pas exploré la sauvegarde des fichiers en profondeur. Le moins qu'on aurait pu faire est sauvegarder les données en fichier .txt sans donner le choix à l'utilisateur de le mettre dans un emplacement précis. Le mieux qu'on aurait pu faire est le scénario indiqué dans ce manuel.

également de devenir portable. Excepté de la plaque arduino, les capteurs de force sont eux aussi très fragile en raison de leurs petites tailles, soit une épaisseur de moins de 1 mm. Alors, il faut les manipuler avec prudence lors de l'installation au sein d'un piano. Par ailleurs, les fils électriques ainsi que les câbles de démarrage ne sont pas tout à fait soudés au complet. En conséquence, s'ils sont suffisamment tirés, les fils peuvent se détacher et perdre leur connexion. Toutefois, cela ne devrait pas poser de problème s'ils restent dans la boîte et si celle-ci est manipulée avec précaution. En bref, tout le matériel est en théorie durable si les contraintes météorologiques sont bien respectées et il est manipulé avec soin.

4.3 Source de puissance (par ordinateur)

Le produit requiert un ordinateur personnel afin de pouvoir alimenter le circuit et exécuter les différents logiciels nécessaires comme Arduino et MatLab. Pour cette raison, il est important d'avoir un ordinateur capable d'exécuter ces systèmes à coup sûr. Comme mentionné ci-haut, il est important de garder le portable à un endroit où il ne réchauffera pas, donc dans une pièce à température de la pièce. Un ordinateur portable idéal aura un port USB et un minimum de 8Go de RAM et 256Go de SSD avec la mise à jour la plus récente.

5 Mesures de sécurité

- Éviter tout contact avec des liquides
- Éviter les fils exposés en raison d'un soudage incomplet
- Veiller sur la surchauffe
- Produit n'est pas destiné à être vendu
- Éviter de bloquer ou de restreindre les ventilateur du portable
- A utiliser dans un environnement sain, à température de la pièce, soit 20°C
- Veuillez réutiliser, recycler ou disposer avec soin une fois terminé avec l'appareil
- Ne peut être utilisé qu'avec de la technologie informatique
- Déconnecter l'appareil avant le déplacement

6 Conclusions et recommandation pour les travaux futurs

Afin de pouvoir améliorer notre prototype finale, plusieurs travaux futurs peuvent être effectués. Ces améliorations seront appliquées suite à la rétroaction du public après avoir leur présenter le dispositif complété. À vrai dire, chaque modification apportée au produit sera en réponse à des critiques faite par les membres du public ainsi qu'à nos propres observations. Dans un premier lieu, le produit est branché à un ordinateur par fil USB de sorte que ce dernier agit comme source

d'énergie qui agit également comme un moyen d'afficher les résultats fournis par l'appareil. Ainsi, afin de se débarrasser complètement de ce fil, on peut favoriser la connectivité par moyen de Bluetooth. Toutefois, ce moyen de branchement nous oblige à trouver une autre source d'alimentation pour l'appareil. Malgré cela, ces nouveaux éléments nous permettront de faire fonctionner le dispositif à l'intérieur du piano à partir d'un ordinateur sans avoir à se soucier des câbles. Deuxièmement, l'appareil fonctionne à partir d'un arduino nano qui possède un bouton marche très minuscule en son centre. De plus, notre produit a été dissimulé dans une boîte de sorte que le bouton est maintenant inaccessible. En conséquence, il n'y a aucun moyen d'atteindre ce bouton et donc il n'y a aucune façon d'allumer ni d'éteindre manuellement l'appareil. Cela étant dit, nous devons chercher un moyen d'accéder à ce bouton soit en modifiant la boîte ou en trouvant un moyen d'étendre le bouton pour qu'il puisse être hors de la boîte que nous avons déjà.

Pour conclure, avec ce document, on a pu résumer notre projet de conception dans son ensemble. Ce document détaillé présente un manuel d'utilisation qui décrit efficacement comment le produit devrait être utilisé afin d'atteindre les résultats souhaités. Ce livrable traite également les éléments clé relatifs à la fabrication et la maintenance nécessaire pour assurer le fonctionnement continue du produit. Par ailleurs, tout au long de ce projet, plusieurs leçons de vie ont été tirées. L'une de ces leçons est le fait qu'une équipe s'additionne pour être supérieure à la somme de leurs membres individuels. D'une manière générale, le processus de réalisation de ce projet semble désorganisé. C'est à cet instant que le travail d'équipe est crucial pour la réalisation de ce dernier. De plus, chaque prototype nous a enseignés à travailler dans les contraintes et les critères donnés, tel qu'un budget. En bref, ce livrable permet de décrire en détail notre produit final tout en élaborant le trajet qui a mené à sa complétion.

7 Bibliographie et Appendice

7.1 Bibliographie

GOVINDARAJAN, Madhu. «Plotting Live Data from a Temperature Sensor Using Arduino and MATLAB», *MathWorks*, [En ligne].

[<https://www.mathworks.com/videos/plotting-live-data-of-a-temperature-sensor-using-arduino-and-matlab-121317.html>] (Consulté le 3 mars 2020)

7.2 APPENDICES

Appendice I: Fichiers de conception

Tous les fichiers de conception sont disponibles sur la page MakerRepo du projet. Celle-ci peut être accessible avec ce lien:

<https://makerepo.com/mariomoubarak1/gng-1503-capteur-de-pression>.

Les fichiers y étant présentés incluent les présentations et les livrables rédigés au long du semestre.