Deseño e implementación dunha gaita MIDI sen fíos en tempo real empregando software/hardware libre

Alejo Pacín Jul

Setembro 2018

fic facultade de informática da coruña

UNIVERSIDADE DA CORUÑA

GTEC

Licenza

fic facultade de informática da coruña

UNIVERSIDADE DA CORUÑA

GTEC

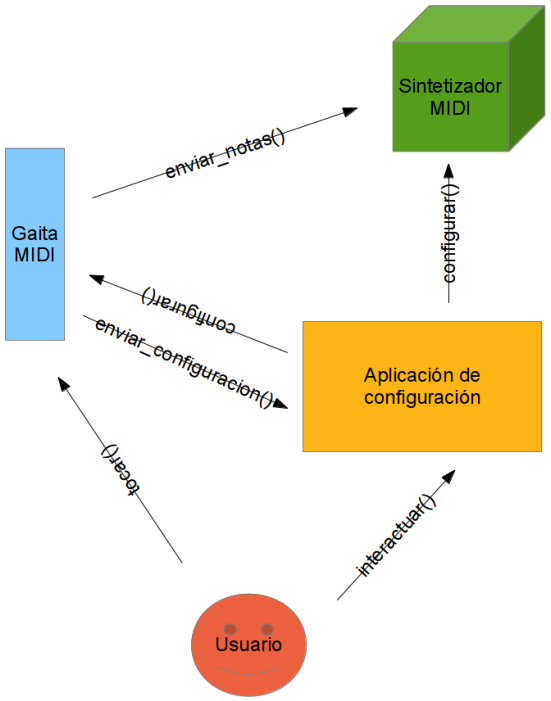
Introdución

- Obxecto
  - Simulación o máis fielmente posible dunha gaita galega
  - Recolleita e realización das esixencias do usuario
  - Análise do mercado
- Obxectivos principais
  - Sen fíos
  - Tempo real
  - Hardware e software libre
- Outros obxectivos
  - Estudo de viabilidade do mercado
  - Estudo de viabilidade das tecnoloxías
  - Estudo de viabilidade do software/hardware libre

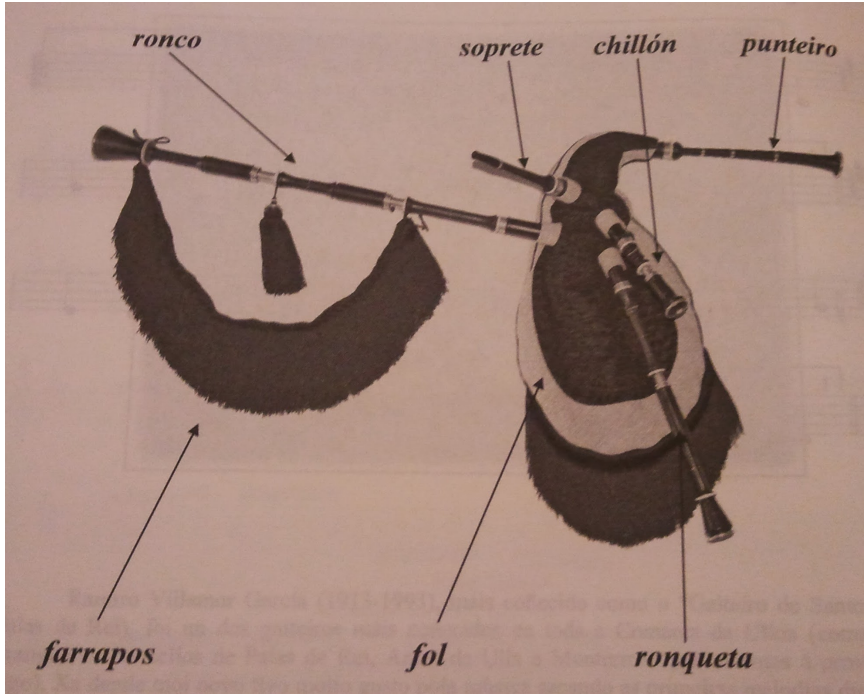
Introdución

- Motivación
  - Mistura de dous mundos académicos
  - Eivas constatadas no mercado MIDI
  - Problemáticas da gaita tradicional:
    - Ensaios no domicilio particular
    - Transcrición de melodías
    - Afinación en condicións desfavorables
    - Interpretación en varias tonalidades
    - Interacción automatizada con outros dispositivos
  - Conxunción de múltiples ramas da EI
  - Fomento da cultura libre

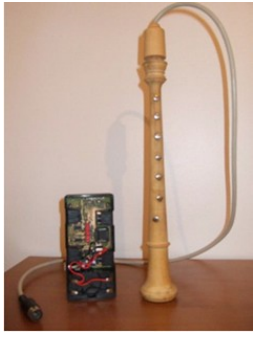
Visión xeral do sistema



Contextualización



Estado da arte



MIDI

- Musical Instrument Digital Interface
- Protocolo serie asíncrono dirixido por eventos que xeran mensaxes MIDI
- Mensaxe MIDI
  - Un byte de estado
  - Seguido de un ou máis bytes de datos
  - Exemplo:
    - 1001xxxx (Note on)
    - 00111100 (Valor 60 que corresponde á nota Do central ”C3”)
    - 0xxxxxxx (A velocidade ou intensidade co que se foi apretada a tecla)

Estudo de viabilidade

- Enquisa:
  - http://enquisa.proxecto-puding.org
  - Extracción de requisitos técnicos e económicos
  - Perfil do usario e coñecemento da competencia
    - Menor de 30 anos
    - Galego residente fóra das capitais
    - Gaiteiro medio con instrumento de seu
    - Predilección por preservar o tradicional con melloras tecnolóxicas
    - Coñece o concepto de gaita MIDI e algún modelo concreto e ten interese en mercar se o prezo é axustado
  - Favorable si, pero significativa?
    - Estimación da poboación total?
      - Problema de Fermi: 1500 gaiteiros
    - Tamaño da mostra necesaria?
      - Cálculo do tamaño mostral coñecido o tamaño da poboación: 90 gaiteiros (6% da poboación)
        -
    - Non significativa (mostra 10 < 90)

$$n = \frac{N}{1 + \frac{e^2(N-1)}{(z^2)pq}}$$

- Nivel de confianza do 95 %.
- $N = 1500$.
- $e = 0{,}1$.
- $z = 1{,}96$ ($\alpha = 0{,}05$).
- $pq = 0{,}25$.

Entón:

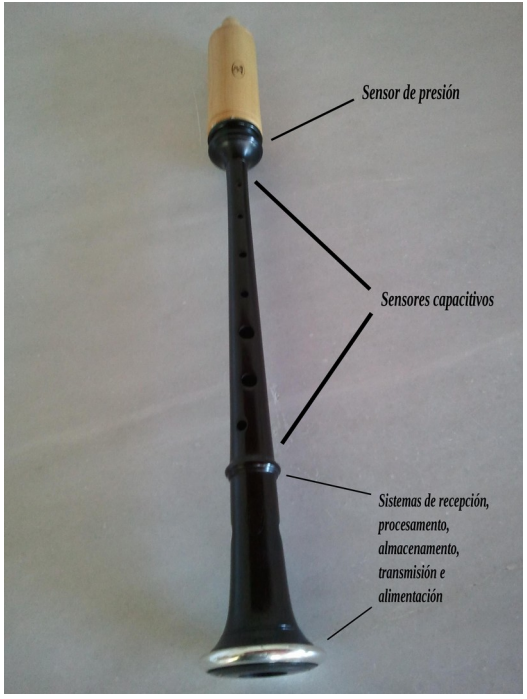
$$n = \frac{1500}{1 + \frac{0{,}1^2(1500-1)}{(1{,}96^2)0{,}25}} \approx 90$$

fic facultade de informática da coruña

UNIVERSIDADE DA CORUÑA

GTEC

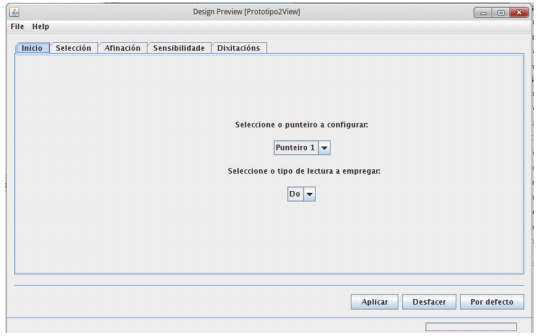
Primeiro prototipo

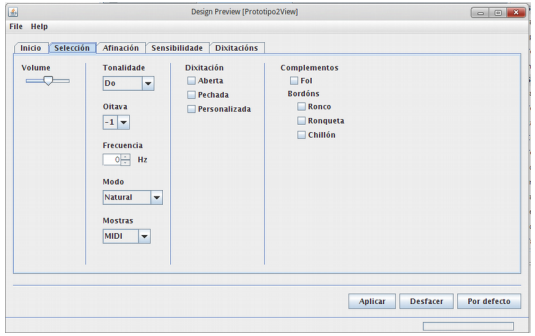


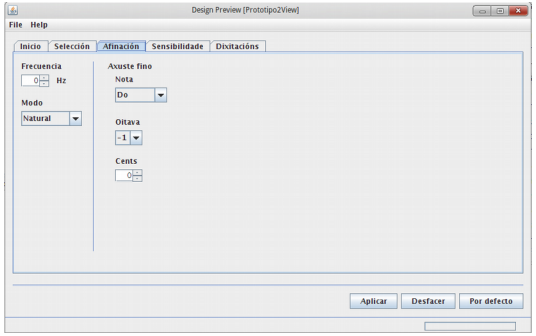
Análise de requisitos

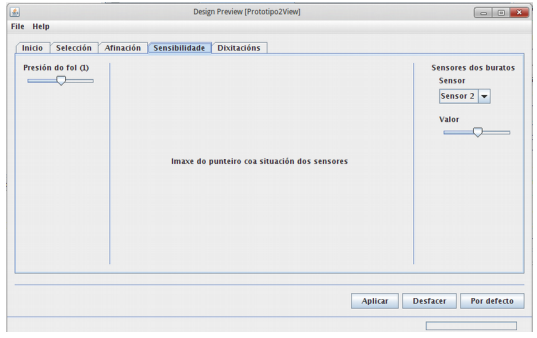
- Extracción de requisitos:
  - Por parte do proxectando
    - Coñecementos propios
  - Por parte de expertos
    - Sobre prototipos
  - Por parte dos usuarios
    - Enquisa

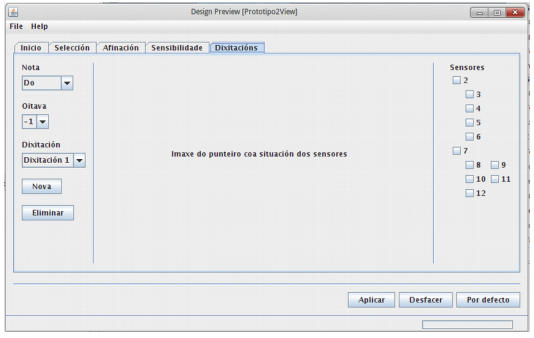
Segundo prototipo











fic facultade de informática da coruña

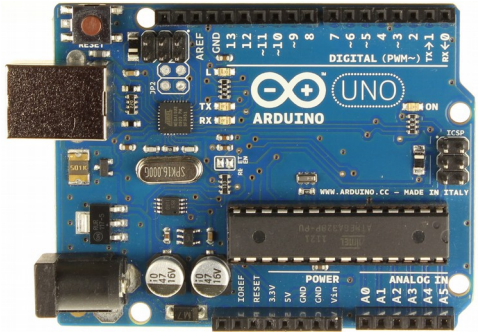
UNIVERSIDADE DA CORUÑA

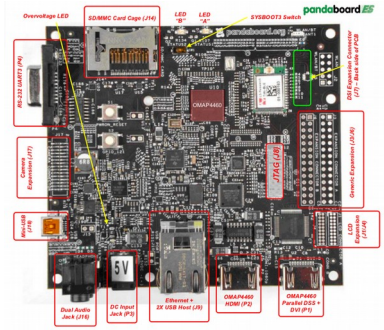
GTEC

Deseño do sistema

- Hardware
- Software

Hardware avaliado



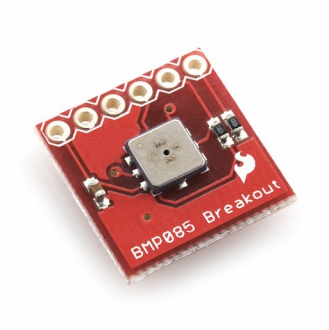


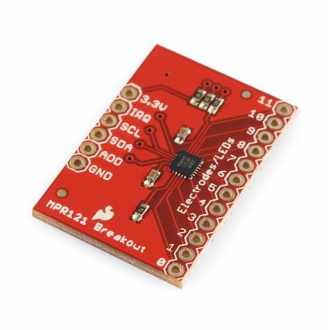
facultade de informática da coruña

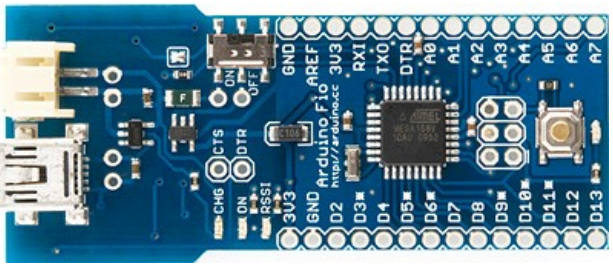
UNIVERSIDADE DA CORUÑA

GTEC

Hardware avaliado





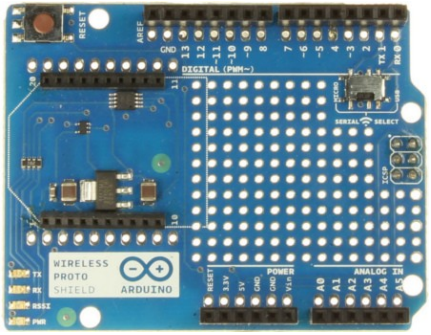


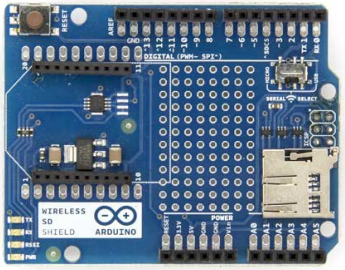
fic facultade de informática da coruña

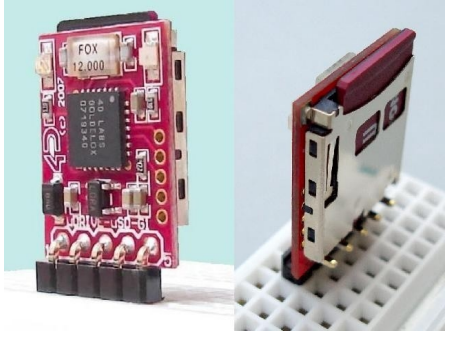
UNIVERSIDADE DA CORUÑA

GTEC

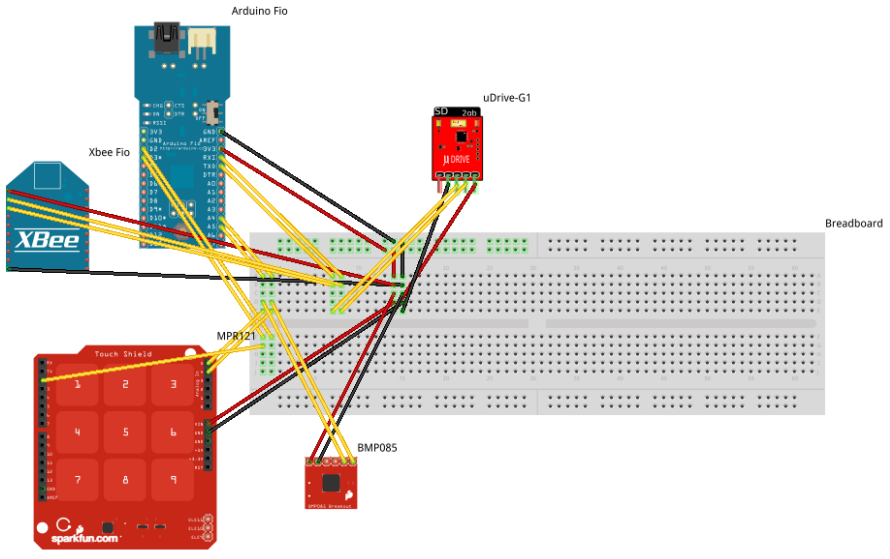
Hardware avaliado

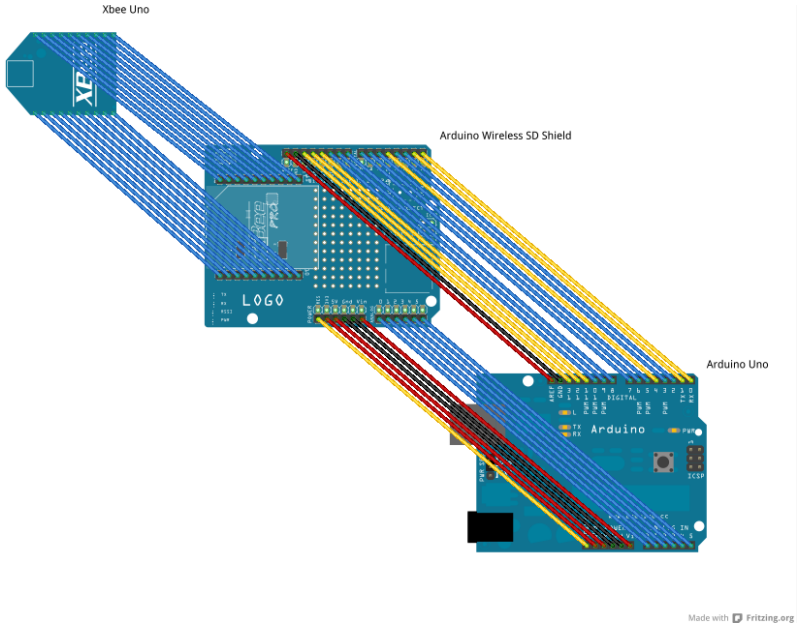




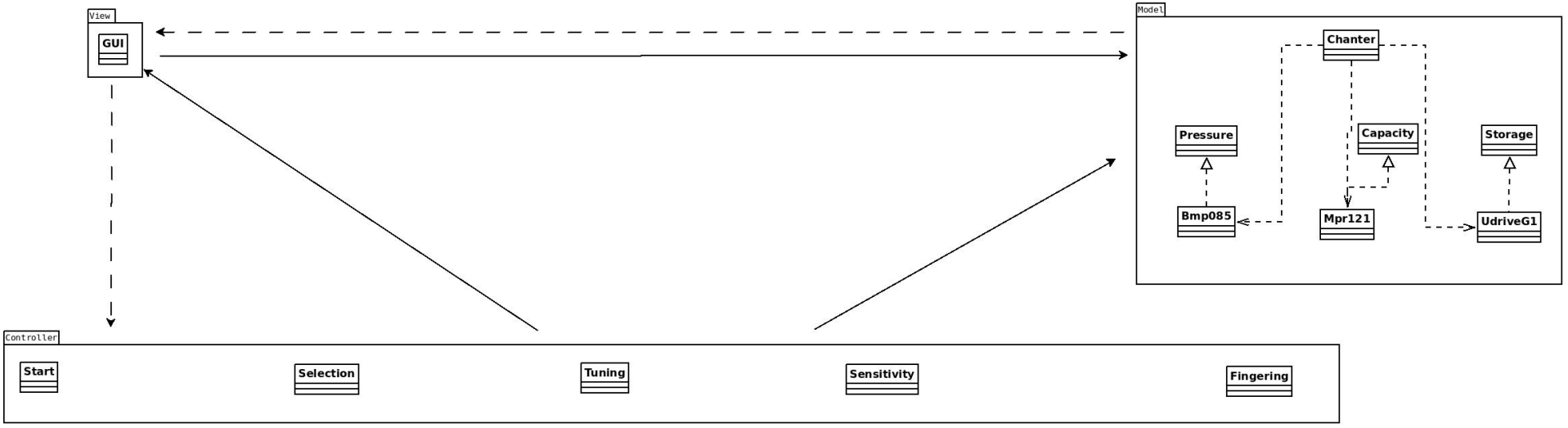


Deseño hardware





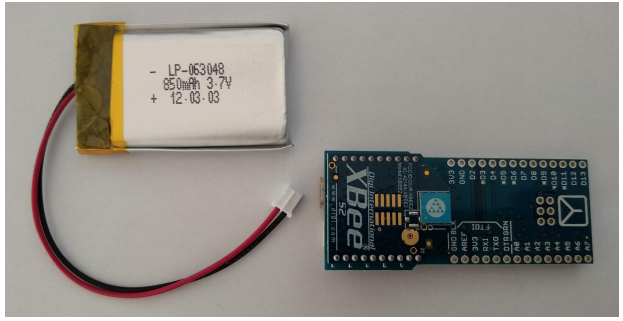
Deseño software

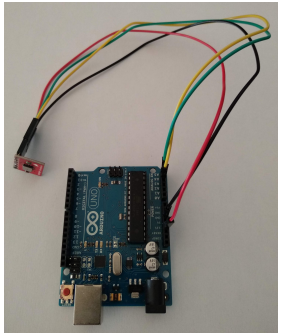


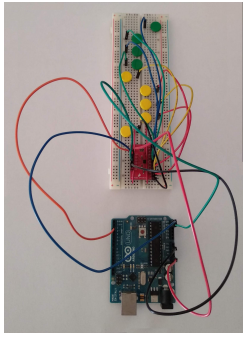
Desenvolvemento do sistema

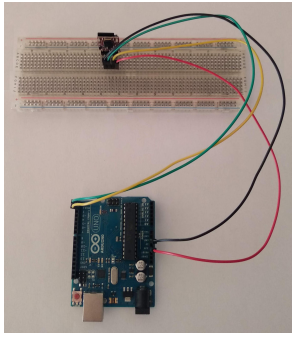
- Prototipo operacional
  - Hardware
  - Software
- Deseño detallado
- Ensamblado e codificación
- Probas (unidade, integración e aceptación)
- Documentación

Prototipo hardware operacional

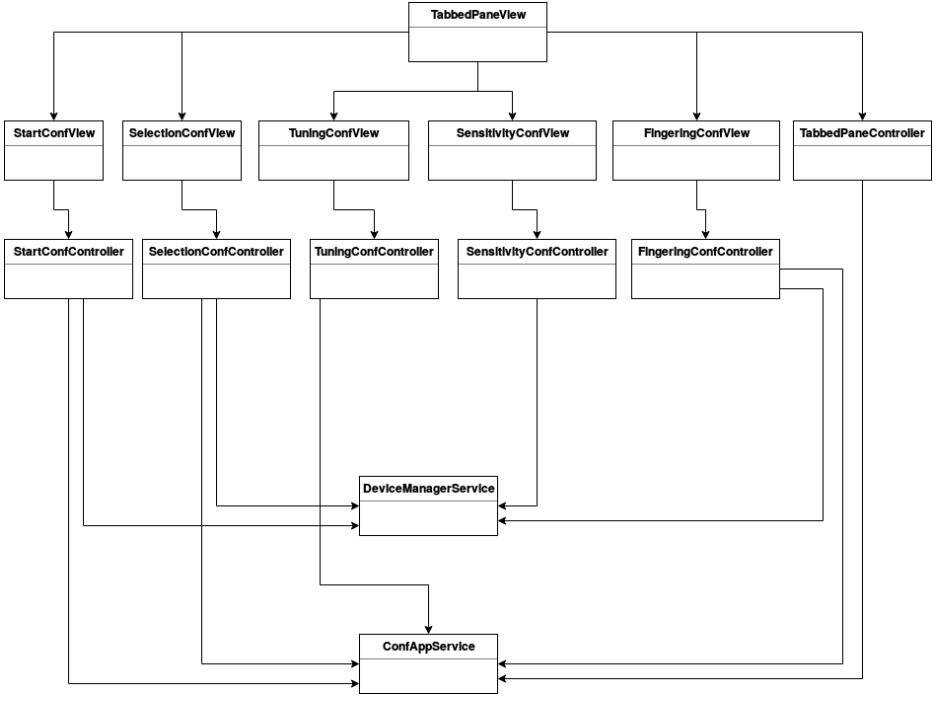








Prototipo software operacional

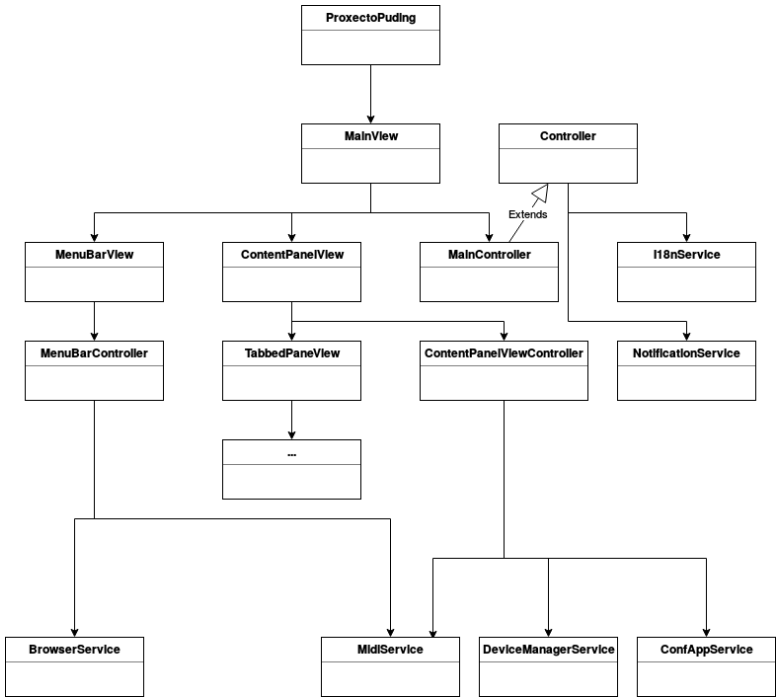


fic facultade de informática da coruña

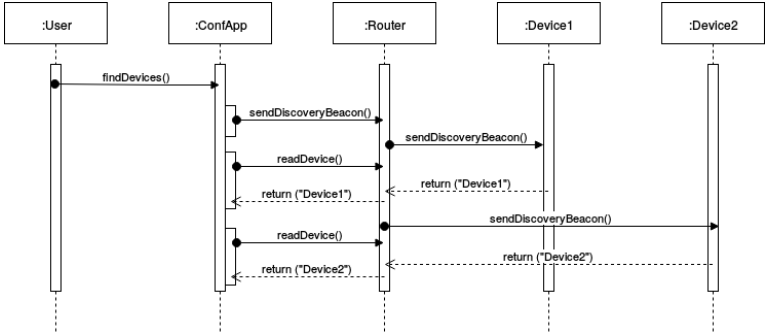
UNIVERSIDADE DA CORUÑA

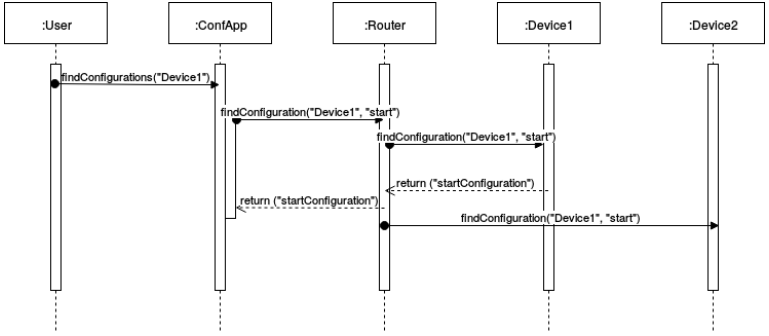
GTEC

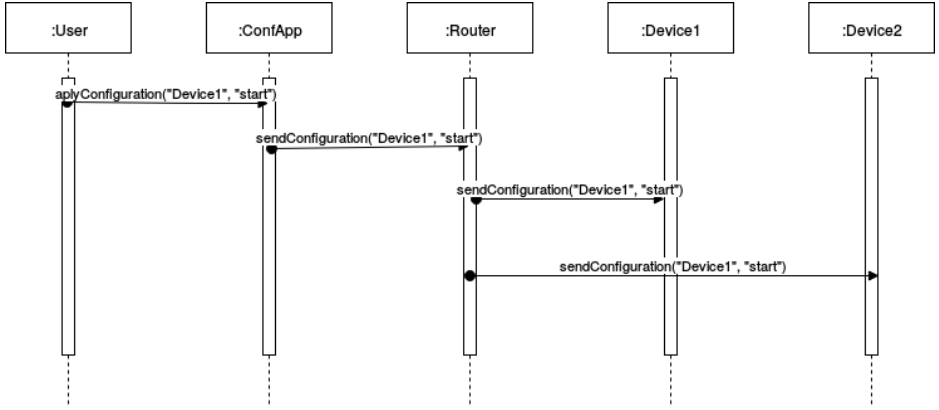
Deseño detallado



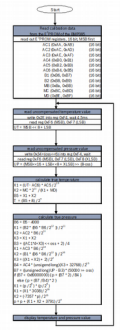
Deseño detallado

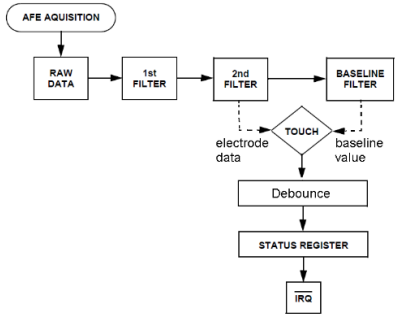




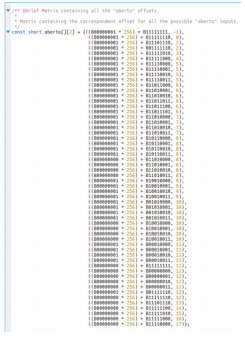


Ensamblado e codificación






```json
{
  "volume": 100,
  "fingeringTypes": [
    true,
    false,
    false
  ],
  "bagEnabled": true,
  "dronesEnabled": [
    true,
    false,
    false
  ],
  "productId": "A000000001",
  "action": "new",
  "type": "SELECT"
}
```



```
/** @brief Matrix containing all the "pechado" offsets.
 *
 * Matrix containing the correspondent offset for all the possible "pechado" inputs.
 */
const short pechado[][2] = {((B00000001 * 256) + B11111111, -1),
                            ((B00000001 * 256) + B11111110, 0),
                            ((B00000001 * 256) + B11101110, 1),
                            ((B00000001 * 256) + B11111100, 2),
                            ((B00000001 * 256) + B11111010, 3),
                            ((B00000001 * 256) + B11111011, 4),
                            ((B00000001 * 256) + B11110110, 5),
                            ((B00000001 * 256) + B11011010, 6),
                            ((B00000001 * 256) + B11011110, 7),
                            ((B00000001 * 256) + B10110110, 8),
                            ((B00000000 * 256) + B11011110, 8),
                            ((B00000001 * 256) + B10011110, 9),
                            ((B00000001 * 256) + B01011110, 10),
                            ((B00000000 * 256) + B10011110, 10),
                            ((B00000001 * 256) + B00011110, 11),
                            ((B00000000 * 256) + B11111111, 11),
                            ((B00000001 * 256) + B01111110, 12),
                            ((B00000000 * 256) + B11111110, 12),
                            ((B00000001 * 256) + B01101110, 13),
                            ((B00000000 * 256) + B11101110, 13),
                            ((B00000000 * 256) + B11111100, 14),
                            ((B00000000 * 256) + B11111010, 15),
                            ((B00000000 * 256) + B11111000, 16),
                            ((B00000000 * 256) + B11110000, 17)};
```

Ensamblado e codificación

MidiServerUnix.java

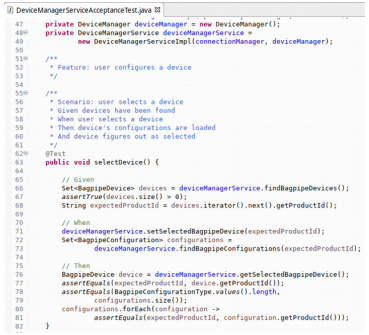
```java
package org.proxectopuding.gui.model.entities.midiServer;

import java.io.File;

public class MidiServerUnix implements MidiServer {

    private static final Logger LOGGER = Logger.getLogger(MidiServerUnix.

    private static final String REAL_SAMPLES = "--config-file=";
    private static final String FREQ_TABLE = "--freq-table=";
    private static final String VIBRATO = "--vibrato";
    private static final String CHORUS = "--chorus=n";
    private static final String DEF_CONFIG_FILE_PATH =
            "/etc/timidity/timidity.cfg";
    private static final String EXECUTABLE_PATH = "/usr/bin/timidity";
    private static final String TEMP_CONFIG_FILE_PATH = "/tmp";

    private final MidiServerGeneral midiServer;
    private String configFilePath;
    private String tablePath;

    public MidiServerUnix(MidiServerGeneral midiServer) {

        this.midiServer = midiServer;
        this.midiServer.setTempConfigFilePath(TEMP_CONFIG_FILE_PATH);
    }

    @Override
    public List<String> getCommand() {

        List<String> command = new ArrayList<String>();

        // Executable.
        command.add(EXECUTABLE_PATH);

        if (getConfiguration() != null) {
            // Real samples.
            if (getConfiguration().useRealSamples()) {
                configFilePath = getRealSamplesConfigFilePath();
                if (configFilePath != null) {
                    command.add(REAL_SAMPLES + configFilePath);
                }
            }

            // Tuning mode, tuning frequency and precise tunings.
            tablePath = getFrequencyTablePath(getConfiguration());
            if (tablePath != null) {
                command.add(FREQ_TABLE + tablePath);
            }

            // Continuous vibrato.
            if (getConfiguration().useContinuousVibrato()) {
                // Only for hardware not supporting manual vibrato.
                command.add(VIBRATO);
                command.add(CHORUS);
            }
        }

        return command;
    }
```

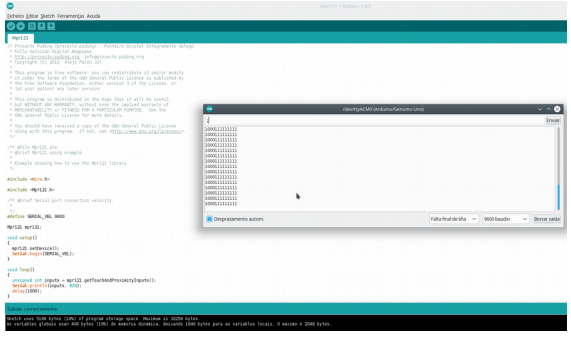
MidiUtils.java

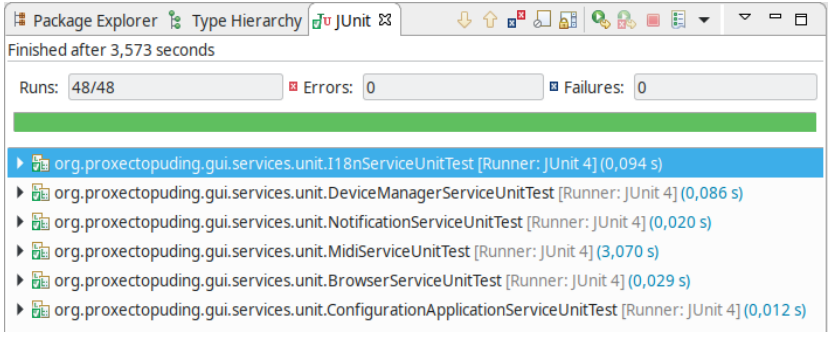
```java
package org.proxectopuding.gui.model.utils;

import java.io.File;

public class MidiUtils {

    private static final Logger LOGGER = Logger.getLogger(MidiUtils.class.getName());

    private static final String FREQ_TABLE = "freqtable.tbl";
    private static double[] pureIntonationRatios =
        {1.0/1, 16.0/15, 8.0/7, 6.0/5, 5.0/4, 4.0/3, 16.0/11, 3.0/2, 8.0/5,
            5.0/3, 7.0/4, 15.0/8};

    /**
     * Generate a table of tuning frequencies to be used by the MIDI server.
     * @param tone Base tone.
     * @param octave Base octave.
     * @param frequency Base frequency.
     * @param usePureIntonation Indicate if it is going to be used pure/just or
     * tempered intonation.
     * @param preciseTunings List of precise custom tunings for particular
     * notes.
     * @param path Directory path where the table is going to be stored.
     * @return The path to the generated table file.
     */
    public String generateFrequencyTable(int tone, int octave,
            int frequency, boolean usePureIntonation,
            Set<PreciseTuning> preciseTunings, String path) {
```

Probas

- Unidade, integración e aceptación
- Metodoloxías TDD/BDD
- 200 tests







fic facultade de informática da coruña

UNIVERSIDADE DA CORUÑA

GTEC

Demostración

Conclusións

- *“Os enxeñeiros somos como navallas suízas, valemos un pouco para todo aínda que non fagamos nada á perfección, pero a pouco que nos esforzamos, brillamos como ferramentas especializadas”*
- *Obxectivos principais*
  - *Sen fíos*
  - *Tempo real*
  - *Hardware e software libre*
- *Outros obxectivos*
  - *Estudo de viabilidade do mercado*
  - *Estudo de viabilidade das tecnoloxías*
  - *Estudo de viabilidade do software/hardware libre*
- *Cuantitativamente*
  - *Código: 18000 liñas*
  - *Tempo: 966 horas*
  - *Documentación: 400 páxinas*

Traballo futuro

- Lector de tarxetas
- Memoria
- Actualización do hardware
- Actualización da interface (Swing → JavaFX)
- Proxectando, doutorando, comunidade?

Traballo futuro

- Proxecto Puding
  - PUnteiro Dixital Integramente Galego
  - http://proxecto-puding.org

Agradecementos

- *“A aqueles que manteñen viva a nosa cultura e, en especial, ós que promoven que dita cultura sexa libre”*

fic facultade de informática da coruña

UNIVERSIDADE DA CORUÑA

GTEC

Deseño e implementación dunha gaita MIDI sen fíos en tempo real empregando software/hardware libre

Alejo Pacín Jul

Setembro 2018

fic facultade de informática da coruña

UNIVERSIDADE DA CORUÑA

GTEC