**Melody Note Final Report**

官佳薇 何青蓉 李世超

**一 项目介绍**

灵感稍纵即逝，本项目的目标是能够记录下一段小调，以音频形式输入，读取识别其曲调，并制成谱子，最终以钢琴弹奏的形式输出，依此将一些日常生活中的小灵感保存起来，以便日后回忆甚至再创作。

任务划分：

①识别音频生成乐谱（官佳薇）；

②根据乐谱弹奏钢琴曲（何青蓉）；

③制作用户友好界面，实现音频获取、上传、播放等功能（李世超）。

**二 音频转乐谱部分**

1. **音频降噪**

使用谱减法对音频降噪处理。根据输入音频计算出噪声频谱，从原始音频中减去噪声。

1. **端点检测**

借助librosa库，进行初步的音频端点检测，返回每个端点的起始时间，对音频做切割。

1. **音符映射**

针对切割后的每一小段音频，计算对应的音符，主要处理流程为：

(1) 给定音频长度阈值，过滤时间较短的音符，避免噪声过大。

1. 对音频进行快速傅里叶变换(FFT)，得到不同频率对应的强度，进行频率筛选：
2. 强度阈值

针对不同音频强弱的差异性，强度阈值取值依据音频整体的强度给定，即根据不同音频强度特征调整强度阈值，以提高准确性。在FFT得到的一组频率中获取最高强度值，参考资料及运行效果，取最高强度值的平方乘0.1作为强度阈值效果较好。

1. 频率阈值

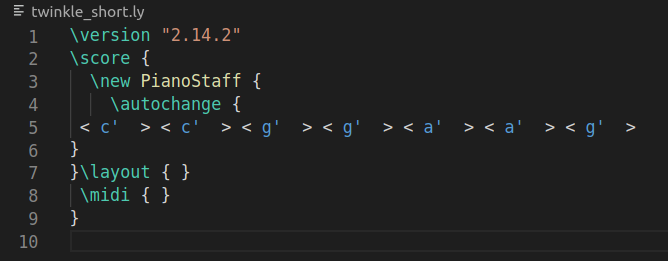
根据人耳听觉特性，设定统一的最高及最低频率阈值，对FFT得到的频率组 合进行裁剪。

1. 获取关键频率

首先将FFT得到的频率进行聚类，将相近频率合成一族，降低噪声。通过丢弃谐波获取音频中主要的频率值。由于本项目多数以哼唱为输入，为简化计算任务，取基频F0作为该段音频中提取的音符。

1. 根据函数映射换算midi库支持的音符值
2. **Lilypond打谱**

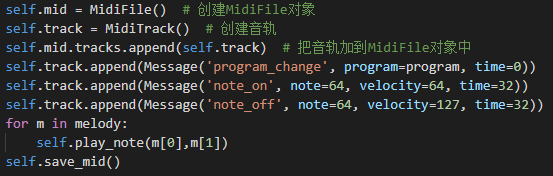
例：输入小星星音频，输出音符及乐谱。





**三 乐谱转钢琴曲部分**

mido编曲流程：



**1. 创建一个MidiFile对象；**

**2. 创建一个（或多个）MidiTrack对象，并将其append到MidiFile中；**

**3. 向MidiTrack对象内添加Message对象**（包括program\_change、note\_on、note\_off等）**和MetaMessage对象**（用以表示MIDI文件的节拍、速度、调式等属性）；

·program\_change：用于更改不同channel的乐器音色

channel：音轨（0-15）（可以更改不同音轨的乐器）

program：乐器编号

·note\_on：音符的开始

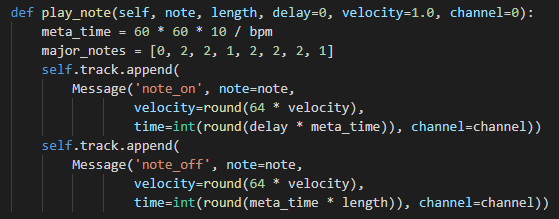
note（0-127）：音符的高低，60代表音高C4

velocity（0-127）：音强，默认为64

time（单位：ms）：该音符写在前一个音符结束多久之后

·note\_off：音符的结束

**4. 向音轨track中添加音符与节拍长度；**



·meta\_time：根据bpm计算出每个节拍的时间长度

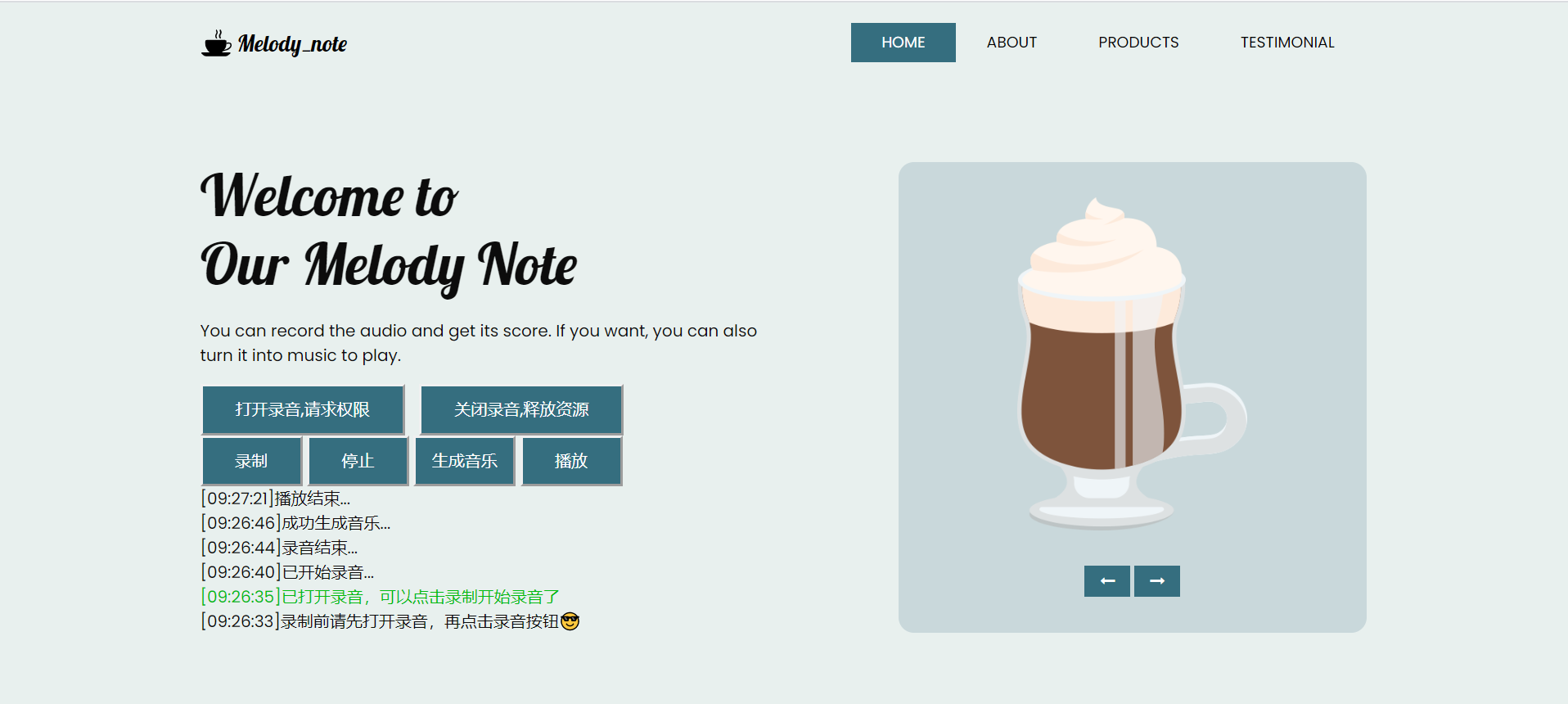
·velocity（0-2）：以64为基准来比较

**5. 保存MidiFile对象；**

**6. 利用pygame播放midi文件；**

（7、将mid文件转换为wav：需要java环境）

**四 界面设计**



1. **通过js与后端python函数交互**

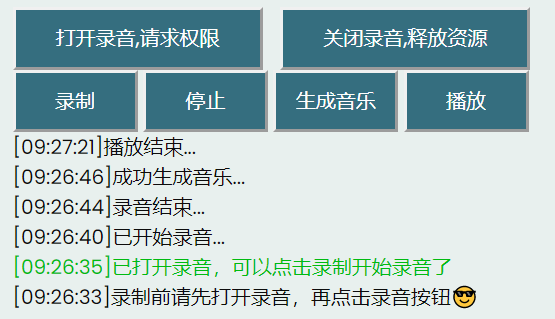
**eg. 利用js实现获取录音权限**



1. **获取录音权限，不同按钮分别调用record录音函数、get\_music转音乐函数、play播放函数**



1. **实时反馈信息**



**4. 自动保存wav文件、note文件（pdf格式）、mid文件**



**五 使用手册**

1. python manage.py runserver 0.0.0.0:8080运行django项目后打开<http://127.0.0.1:8080/index/>;
2. 点击“打开录音”按钮，获得录音权限；
3. 点击“录制”按钮，开始录音，“停止”按钮停止录音；
4. 点击“生成音乐”，需要稍等一下哦，可在后台查看是否转换完毕；
5. 点击“播放”按钮即可听到钢琴曲；
6. 有关保存文件请查看work文件夹下的说明.txt。