SCANIANO2: OCR/OMR QR CODE GENERATOR

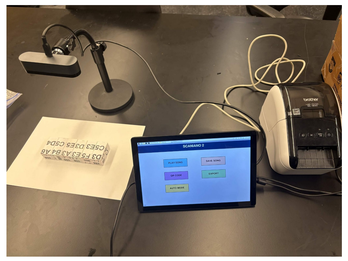
STUDENTS: YUMING CHEN, JIMENA GUZMAN, LUKE MILLER, DECLAN NEWELL, RIYA URS

Problem Statement

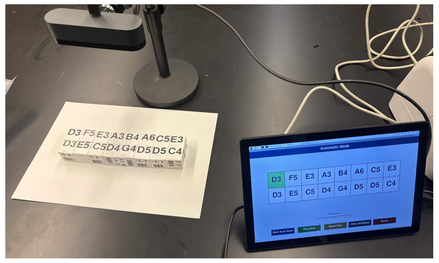
- The objective of Scaniano2 is to improve Scaniano1 so that users of all ages can more easily create and play melodies without the need for music or instrument training. Improve the QR encoding process so as to more easily capture, store and play songs by scanning note blocks, song books, sheet music and URL webscrapes. Print QR labels for Songbooks.
- Implement a touchscreen tablet form factor to replace the "boom box" concept and add a midi out capability that can drive a music synthesizer.
- Use camera-based scanning and image processing (OCR and OMR) to enable an "automatic" mode that reduces the UX burden and enhances the fun !

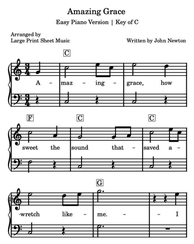
Hardware Integration

- Runs on a Raspberry Pi Model 5, with an integrated TouchScreen
- The camera and label printer are attached via USB to the Raspberry Pi
- QR code scanners are connected via USB receivers
- Magnetic cubes labeled with note names and printed sheet music are scanned by the camera, while QR codes are handled by the QR code scanner



Optical Character/Music Recognition





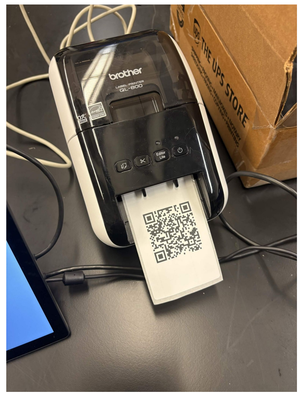
- Optical Character Recognition (OCR) processes an image of magnetic note cubes and converts them to text using the PyTesseract library
- Optical Music Recognition (OMR) of sheet music was implemented using the Python-based Oemer library, which performs automated music notation recognition and extracts musical note information from the score
- Songs are saved and stored as JSON files containing the title, key, and notes as text, and can be exported as a QR code, MusicXML, or MIDI file
- Automatic Mode, as seen in the image above, condenses the OCR into a quick Scan -> Play experience, providing visuals and testing for songs prior to saving

Webscraping

- The user inputs an html file that contains tablature for Harmonica music (primarily from Harptabs.com consistent with Fair Use laws)
- The html file is scanned and processed, then converted into a .json file
- Goal: to create a dynamic webscraping pipeline for users to search for any songs they would like to add automatically

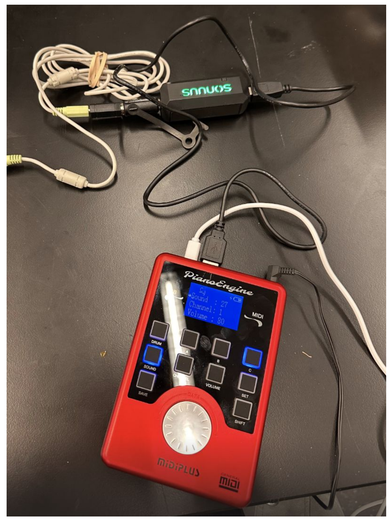
QR Code Generation

- After an image of the music is captured, note data is exported into MusicXML
- Provides a structured digital representation of the musical notation
- MusicXML data is then converted into a compact string representation for encoding
- QR code generation was implemented using the Python Pillow QR library, which converted the encoded musical note string into a scannable QR code
- The generated QR code is then sent to the Brother label printer for physical printing



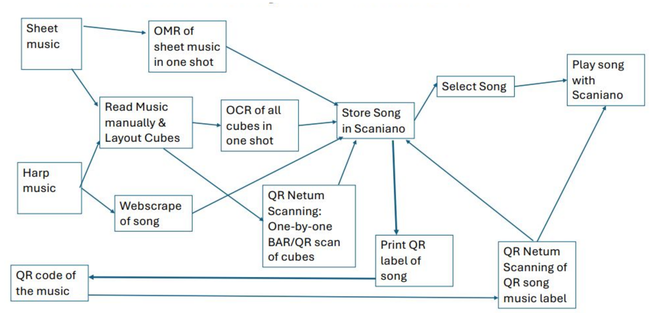
MIDI Connections

- The audio output from the touchscreen interface is routed through the headphone jack and converted into MIDI data using the Sonuus audio-to-MIDI converter
- The generated MIDI signals are transmitted to the PianoEngine sound module
- Performs real-time sound synthesis based on the detected musical input
- The PianoEngine module outputs the synthesized audio signal to an external speaker
- Provides high-quality audio playback suitable for demonstration and live interaction
- This audio chain provides a compact pipeline for converting analog audio input into MIDI-controlled output
- MIDI architecture allows flexible integration between embedded hardware and audio processing components

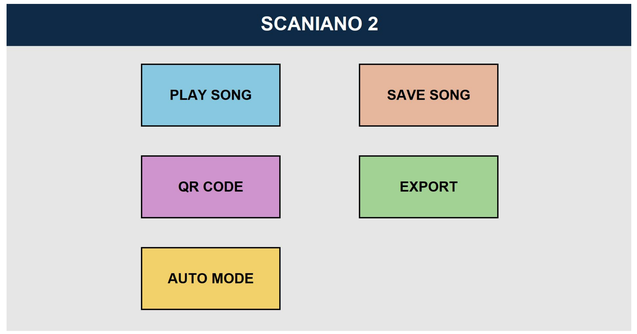


User Interface/Workflow

- System Configuration



- User Interface



- Simple and intuitive
- Touchscreen
- Functionality without Friction

Future Work, References, and Acknowledgments

- Port Scaniano2 functionality into iOS application
- Develop a dedicated MIDI hardware interface capable of converting Raspberry Pi 5 audio output signals into MIDI data and transmitting them through a serial communication interface

Faculty: Dr David B Laning, Affiliate Professor ECE
Graduate Students: Ben Davis
Testing Volunteers: Thomas & Kappy O'Brien, Dorian & Elena Varga

ELECTRICAL & COMPUTER ENGINEERING
UNIVERSITY of WASHINGTON

ADVISER: DR DAVID B LANING, AFFILIATE PROFESSOR ECE
SPONSOR: DR DAVID B LANING, AFFILITATE PROFESSOR ECE