Main Goal

To create a software and hardware system that can transcribe full songs (multi-instrument audio) into sheet music or MIDI files, allowing users to choose specific instruments (e.g., piano, drums, vocals) to transcribe and making transcription more accessible.

Key Objectives

Multi-Instrument Transcription:

Use pretrained AI models (e.g., Magenta’s Onsets and Frames for piano, DrumRNN for drums) and source separation tools (e.g., Demucs or Spleeter) to isolate individual instruments for transcription.

User-Specified Instrument Transcription:

Allow users to specify which instrument they want to transcribe from a song (e.g., extract and transcribe only the piano or drums).

MIDI and Sheet Music Generation:

Convert the transcription results into MIDI files for playback or into sheet music using libraries like Music21 or external tools like MuseScore.

Real-Time Processing (Optional/Stretch Goal):

Explore real-time transcription for live audio input, focusing on low-latency processing.

Hardware Integration:

Deploy the system on a Raspberry Pi or similar hardware to create a portable device capable of processing pre-recorded songs or live performances.

Customizable and User-Friendly:

Provide an interface or command-line tool that is easy to use, with options for:

Uploading audio files.

Choosing instruments for transcription.

Viewing or saving the results.

What You’re NOT Doing

Training AI Models from Scratch:

Training models like Onsets and Frames requires massive datasets and computational resources, which are out of scope for a senior project.

Instead, you’ll use pretrained models and focus on adapting and integrating them for your specific use case.

Developing a New Transcription Algorithm:

Your work builds on existing transcription tools and models, but your innovation lies in how you combine and optimize these tools.

Full-Scale Commercial Product:

This is a prototype system to demonstrate the feasibility and functionality of your approach.

What Makes This Project Valuable?

Real-World Application:

Your system can help musicians, hobbyists, and educators transcribe music for learning or practice.

It lowers the barrier for transcription, even for those with limited music theory knowledge.

Engineering Focus:

Demonstrates your ability to integrate machine learning, digital signal processing (DSP), and hardware/software design.

Scope Alignment:

Using pretrained models allows you to focus on solving real challenges like:

Multi-instrument transcription.

Noise handling.

Real-time processing.

Resource constraints on hardware.

Deliverables

Working Software:

A Python-based tool that can process an audio file, separate instrument tracks, and transcribe them into MIDI or sheet music.

Hardware Prototype (Optional):

A Raspberry Pi-based system with a microphone input or audio upload capability, capable of running the transcription software.

Project Report:

A detailed report explaining:

The challenges you addressed.

How you adapted and integrated pretrained models.

The system architecture and its functionality.

Demonstration:

A live or recorded demo showing:

The system processing an audio file (e.g., a popular song).

Outputting transcription for selected instruments.

Visualizing or playing back the results.

Next Steps

Set Up Environment:

Install and test Magenta, TensorFlow, and the source separation tools (e.g., Demucs, Spleeter).

Prototype Source Separation:

Separate a song into individual stems (vocals, piano, drums, etc.).

Integrate Transcription:

Process the separated tracks with transcription models for each instrument.

Focus on Output:

Generate MIDI files or sheet music from the transcription.

Iterate and Refine:

Optimize performance, add a user interface, and explore hardware deployment.