**Overview and Technicals**

The project under consideration, nicknamed SPLALO, is a computer based real-time audio analysis program capable of transcribing live instrumentation as sheet music. During early phases of development and testing, audio analysis will require a user-preset time frame and storage space for audio files. Additionally, SPLALO serves as an assistant for interpreting the transcribed technical notation of sheet music. To successfully utilize the program four I/O devices are necessary; a computer mouse & a keyboard for user command input, a monitor displaying the program’s output messages and a standard microphone.

To effectively run SPLALO one needs to place the microphone within 2 ft. of the instrument of choice and set the length of recording time before clicking the “Record” button. Once the predefined time has elapsed, the program will automatically stop recording and display some message to alert the user of this. It will store the recording as an audio file in some centralized location on the host computer (which can be played back independently from SPLALO). At this stage, the program will access the file and analyze it to determine the various properties necessary for generating sheet music. These properties include (but are not limited to) the overall tempo of the piece, individual note values, individual timing values and individual amplitude values. In the finalized version analysis for these properties will be done while the program is recording the live instrumentation. Once the necessary attributes have been determined, the appropriate sheet music equivalent will be generated and displayed on the screen. For pieces spanning multiple pages, the user will need to scroll through the program’s main display to view the rest of the piece.

To effectively create this program, certain functional and nonfunctional requirements must be met. These are listed and described as follows:

**Functional Requirements**

1. **SPLALO must record audio input and store it as a sound file if necessary**

To achieve this, the program uses the standard Java Sound Library. The library provides several useful tools for managing audio I/O devices and file storage. The tools allow me to record using a selected microphone and store the audio input as a sound file (as mentioned for early development stages). These sound files contain byte values (numbers between 0 and 255) that represent the amplitude values at periodically sampled times. These samples may be stored in mono (where one amplitude sample is noted per sample time) or stereo format (where two distinct amplitude samples are noted per sample time). Each amplitude sample may be represented either by 8 or 16 bits. The program is initially preprogrammed to store them in 16 bit mono format for more accurate yet easily processable results. The final product should take these byte values in, store and analyze them as the program is listening for audio input.

Since the analytical phases that follow require double values (numbers with decimals), the byte values obtained must be converted to their double value equivalents. This requires storage in some array and step-by-step conversion of data types.

1. **The program must accurately determine the timing values of individual notes**

SPLALO must attempt to identify the start and end times for individual notes and rests to accurately represent a piece as sheet music. Two methods were proposed to obtain these values in conjunction with the frequency values. The initial solution involved a “sliding time window”, attaining the frequency values for set intervals of time (typically a few milliseconds), and making timestamps whenever changes were detected in perceived frequencies. This solution proved inefficient when testing on notes from instruments as opposed to pure tones (which are standard sine waves). Hence an alternative was proposed where timestamps will be taken when certain averaged amplitude thresholds (based on the past few time windows) are exceeded; simulating a new note being heard. When this averaged threshold falls below a “rest threshold” the program will also record a timestamp to represent the end of a note/beginning of a rest. This factors in tests on actual notes and rests which are fundamentals in playing music.

1. **The program must accurately determine the note(s) being played.**

To successfully determine the notes being played (or lack thereof) a frequency analysis must be implemented. This deduction will be performed after the music piece has been segmented by rhythm into notes and rests. An algorithm known as a Fourier Transform takes these segmented waveforms and determines strengths of present pure tones varying in frequencies. This is not as simple as finding the frequency of a pure tone since a single note from an instrument comprises of a series of pure tones varying in amplitudes. After further filtering and evaluation the dominant frequencies may be obtained.

1. **The program must determine dynamics based on amplitude values**

Assuming the newly proposed timing algorithm is implemented, boundaries similar to the “rest threshold” may be implemented to determine when a particular section is louder or softer than usual. Tags like (mezzo) forte, (mezzo) piano, crescendo and fortissimo serve as indicators for how the instrumentalist should play segments of music pieces.

1. **SPLALO must perform holistic analysis to deduce tempo and time signatures**

The timing patterns between subsequent notes must be inferred from the previously stored deduced individual time values. This may require less formal mathematical expressions than Fourier transforms and other analysis done by the program; however, sophisticated pattern recognition algorithms are necessary to attain accurate results.

1. **SPLALO must perform holistic analysis to discern the key of the piece**

Can be determined by comparing a series of notes to some predefined formulas for the various scales in music. An example would be the major scale that follows the following formula: tone, tone, semitone, tone, tone, tone, semitone. Knowing this and comparing the notes present in the piece, one may determine “the key of the piece”. One challenge while deducing this characteristic is that notes of a scale need not appear in the exact order to be considered for the scale. Additionally, the root note of the scale may not even be played in the series being analyzed.

1. **SPLALO must accurately transcribe to technical notation**

This would be the stage where the program simply compiles results obtained from all of the previous steps and notates it appropriately.

1. **SPLALO must provide information on understanding technical notation**

This is stated in conjunction with SPLALO’s secondary aim. It will be implemented via some help button located on the GUI and should provide a list of music theory topics.

**Non-Functional Requirements**

1. Program must be able to analyze properties of notes and draw conclusions as quickly as possible (ideally within a second for each note).
2. Program should run independent of host machine.
3. SPLALO must be as accurate as possible in all interpretations. Appropriate timing, frequency and amplitude values must be recognized for accurate interpretation and transcription.

**Business Proposal**

As suggested in the technical description, SPLALO will primarily serve as a tool for transcribing from sources in real time. It eases the user’s process of composing music or playing it; doing so in a convenient and simplified manner. Such a product is typically rare to find or hard to use, and the market of music theory enthusiasts is immense. This consumer group encompasses casual music listeners new to the technicalities, expert instrument players and anyone inbetween. It is capable of taking the market by storm if appropriately advertised through music related channels such as Spotify or YouTube (exclusively on videos related to music theory).

What if one has an interest for the technicalities of music theory yet requires a source for learning such skills? SPLALO is not limited solely to the aforementioned consumers.

This issue is overcome through the secondary purpose of the program. Its use as an assistant for interpreting sheet music attracts a new group of music enthusiasts with limited literacy in music theory. Several tools currently exist to perform this task as is and they come in all forms from Youtube videos to piano simulations. However, such tools are standalone products that typically have a shorter term of use. Users would utilize such products for a limited amount of time and discard them upon completion of the courses or even midway through. I believe this is due to such products’ lack of continuous engagement and appeal to consumers. They present themselves as fixed curriculums and courses whose concepts must be applied separately by the consumer. For this reason some would not consider such products as worth their value and may be disinclined to pay for them.SPLALO, however, presents itself in a different way by attaching this secondary functionality to its core purpose. This implies that music illiterate enthusiast are guided from the basics throughout while applying the concepts to their recorded instrumentation and resulting transcriptions. It captures the consumer from early stages on music theory and generates interest in the subject using the same product.

Due to the tools and nature of the program it should be capable of running on any computer. The choice of programming language allows cross-platform implementation of SPLALO; this implies adjustments need not be made in code to run on different operating systems. This guarantees a reduction in budgets spent on larger teams and, most importantly, time until release. Developing in a singular language means that integration of code segments between teams runs smoother.

Furthermore, the developer of this product is not required to deliver any special hardware with SPLALO. The essential I/O devices are either built into computers or may be purchased by the consumer of their own accord; this narrows the production budget down even further. It also frees from needing to physically release the product and utilizing distributors; hence eliminating the middleman. The product may simply be downloaded from some online source upon payment. Initially this may prove unsafe with the possibility of some backdoor hack however such a threat may be minimized. As most products do when distributed via less safe mediums, product keys may be issued out upon purchase of the product. This will certify the consumer of SPLALO indeed paid for it.

A tool as diverse in installation as SPLALO, if appropriately advertised, is certain to turn a profit. Its low costing production and appeal across the vast music community will surely attract investors.

**Broader Impact**

SPLALO aims to generate and maintain interest in the technicalities of playing and reading music. It strives to simplify the process of transcription; freeing composers’ minds to handle more creative tasks regarding their music. It seeks to guide users towards a greater appreciation for music theory. The music enthusiast community may be connected from the ends of the creator and appreciator. With the increasing use of SPLALO the admiration for creativity through music may be elevated by knowledge on the topic. Such a benefit extends beyond the boundaries of entertainment and art.

Recent neuroscientific studies suggest the brain is more active when processing music. During an average listening session, the brain attempts to separate the multiple elements in a song, identifies them individually with their timing or melodic patterns and reexamines the piece as the sum of parts. These processes occur in a matter of milliseconds and reflect in one’s identification of instruments/voices in a song or recognition of rhythm. The ability to do this serves as a localized workout for certain parts of the brain. Imagine the kind of exercise one puts their brain under when well versed in music theory! Such feats can be attained through the constant use of SPLALO’s interpretation assistant. The deeper understanding of frequency & time patterns for their emotional and psychological effects would push the brain to greater extents. It would further engage the analytical sections; maintaining an active and healthy brain. This would easily translate to non-music settings where one’s analytical skills must be utilized socially or academically. This is why it is believed that the world’s greatest composers are some of the brightest minds to ever live.

Aside from their mastery in the composition and interpretation of music, what set greats like Mozart and Bach apart from the crowd is their proficiency in playing instruments. Simultaneously implementing the previously mentioned skills requires use of the brain’s visual, auditory and motor sections; employing more brain power and gradually improving one’s capabilities. The development of one’s brain may be accelerated through playing instruments and SPLALO aims to assist this with its core transcription functionality.

Such tools are especially useful for younger users practicing an instrument or music theory. As professionals have stated, the early years of brain development are crucial for laying a foundation to last a lifetime. By stimulating a child’s brain one is expanding its capabilities for the future; which is reason enough for parents to try and get their children interested in playing some instrument. SPLALO serves as an engaging guide to music by allowing users to enter some of their favourite music for notation and practice. It encourages regular practice, in turn improving brain functionality in younger users.

One crucial thing to note about such a program is that there are no direct negative influences environmentally and socially. The only possible downside would be overusing the program for practice to the extent of annoying others.