

# Project Proposal

ITS Music

Chase Sprague

## Problem to be Solved

### Description

*A description of the problem to be solved.*

Music classes in school are often underfunded and are given limited resources. Because of this many schools have a single music teacher who is trying to teach 40 or more students, learning a dozen or more different instruments, at a single time. The teachers are supposed to teach instrument fingering, breathing (for wind instruments), music theory, reading notes, sight-reading, and rhythm. This undermines the teacher's ability to teach. It is impossible for them to focus on helping each student grasp their instrument completely.

### Existing Solutions

*A description of existing solutions for that problem, specifically to contextualize why your solution is needed.*

In my research I was able to find very few actual ITS currently available for teaching students to play music. The closest tools I could find to help students are *ultimate-guitar.com*, *A Stylus-Driven Intelligent Tutoring System for Music Education Instruction*, and *SmartMusic*.

*Ultimate-Guitar* is a well known website used for helping students learn to play songs on guitar. It is not an ITS, but it is definitely a good resource for determining what methods might help a learning student. The website provides guitar tabs to the user so that they may learn to play the song. The website also provides a paid tool that will play the song along with the user at an adjustable tempo.

*A Stylus-Driven Intelligent Tutoring System for Music Education Instruction* seems to be an ITS with focus on learning basic music theory. The tool claims to emulate how a music teacher would approach the subject, giving proper feedback when necessary.

The software is called SmartMusic [<https://www.smartmusic.com/>] allows teachers to assign music to practice, records practice sessions for the teacher to evaluate, and provides examples for guided practice. While I am not sure how much I can complete in a single summer semester, these are abilities I would like my application to provide. SmartMusic has no intelligence aspect for it and is designed to be a simple practice tool. I would like to take this idea to the next step and add some form of intelligence to it.

I think one of the main reasons for a lack of tools in the music field is funding. Since the arts are often under valued and are given very little, developers are unlikely to make much money in the creation of tools to benefit the area.

## Tool Design Description

*A description of the design of the tool you will create.*

I am proposing an intelligent tutoring system that would be used along side of a music class to teach the foundational requirements needed to learn how to play an instrument. This would reduce the load on the teacher significantly, and would allow them focus more on teaching proper techniques. This application is being developed for PC environment and would be intended for home use. This application will teach users to read rhythm patterns and the notes on a music staff for both bass and treble clef. Once the user has mastered those, the software will move to teaching scales, chord structures, and mapping notes to an instrument (guitar). When this is completed, the next step will be for the application to start combining rhythms and notes. This entire process will obtain user input by typing and audio. The program will track the user's progress and provide feedback on how to progress through its content.

## Task List

*A task list of the tasks that must be completed to execute and deliver your project. Make sure to include the required tasks as well, such as the intermediate milestones and final paper.*

Pre-proposal (completed):

- Sketch out possible layouts
- Create some basic GUI interfaces
  - Main window where controls will sit
  - Music staff display that is capable of showing notes
- Create a metronome that will be adjustable as students progress
- Create a communication protocol that will work between the backend and frontend of the program.

For Milestone 1:

- Add teaching content

- Add data for music scales and modes (Major (Ionian), Dorian, Phrygian, Lydian, Mixolydian, Minor (Aeolian), Locrian, Chromatic, Blues, etc..)
- Add test rhythm patterns (Approximately 20 introductory rhythms that can be cycled through)
- Add music keys (Ex. Key of C# or Bb)
- Add clef types (bass and treble)
- Note mapping to instrument (guitar)
- Chords (note displays and fingering charts)
- Connect metronome to displayed notes/rhythms
  - Create pointer that moves along music staff at tempo

#### For Milestone 2:

- Textual user input
  - Checking knowledge of reading notes and note types
- Audio user input (still unsure of how to complete)
  - Tuning
  - Checking for correct notes
  - Checking for correct rhythms
- Create feedback dialogs
  - Note was incorrect (sample audio output)
  - Rhythms was incorrect (sample audio output)
  - Scale was incorrect (provide point of mistake)
- Create database to store user progression
  - Store exercises completed
  - Store current tempo for each exercise
  - Store user mistakes
- Provide suggestions on what to complete next
- User testing/feedback

#### For Final Deliverable:

- User testing/feedback
- Complete final paper
- Complete final presentation
- Complete final project

#### Timeline

Week	Week Of	Description
1	05/15/17	Assignment 1/2
2	05/22/17	Assignment 3/4

3	05/29/17	Assignment 5
4	06/05/17	Pre-proposal
5	06/12/17	Sketch possible layouts Create basic GUI Create information protocol Create metronome Proposal
6	06/19/17	Add teaching content Status Check 1
7	06/26/17	Add teaching content Connect Metronome Status Check 2 Milestone 1
8	07/03/17	Textual User Input Audio User Input Status Check 3
9	07/10/17	Create Feedback Dialogs Create Database Provide user suggestions Status Check 4 Milestone 2
10	07/17/17	Developer Testing User Testing Status Check 5 Work on Final project Work on final paper Work on final presentation
11	07/24/17	Complete final Project Complete final Paper Complete final Presentation

## Milestones

*Descriptions and deadlines for your two intermediate milestones.*

## Milestone 1 - July 2nd

Milestone one will be completed on July 2nd and will consist of a functional prototype of teaching content, presented as an executable with necessary dll's. This prototype will contain teaching content for: music scales and modes, rhythm patterns, music keys, clef types, note mapping to guitar, and basic chords. The user will be able to switch between each of these teaching points. The prototype should also have a connected metronome for the content that is played to a rhythm. The goal of this milestone is to put the teaching content into a usable form for a user.

## Milestone 2 - July 16th

Milestone one will be completed on July 16th and will consist of a functional prototype with the intelligence integration, presented as an executable with necessary dll's. This prototype will allow for user input (textual and audio). It will give some form of feedback based on correctness. It will store user progression and give recommendations on what to do next.

## Resources

*A technical description of the tools, languages, and other resources that will be used.*

- I will be developing my software in Visual Studio Professional 2013 IDE.
- I will but using C++ as my language of choice.
- Development is being done on a Windows 10 computer.
- FftGuitarTuner for audio input.
- The class library for ITS references

*A description of the integrations or external resources that will need to be obtained, as well as spring-back plans in case portions of these details cannot be completed.*

As a component to my software, I plan on using FftGuitarTuner. This is free source code available online that allows for using the computer's microphone for sound input. This software is available at <https://www.codeproject.com/Articles/32172/FFT-Guitar-Tuner>. I hope this code will be adequate for my audio input. If this software does not work, I will look into alternative methods for audio input. If that fails, then I will resort to using keyboard input that will simulate the notes to be played.