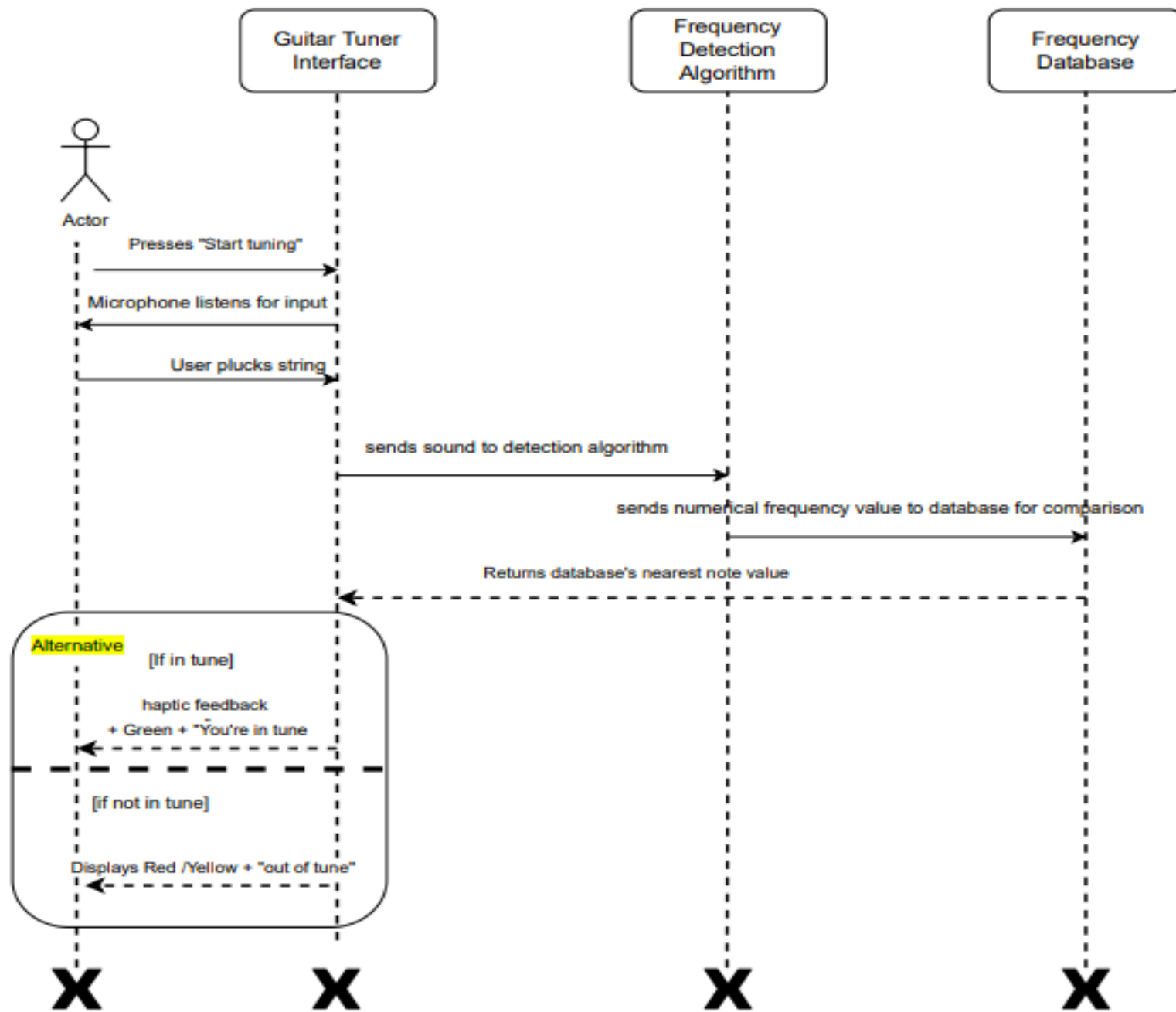


A close-up, artistic photograph of a guitar's neck and strings. The strings are in sharp focus, showing their metallic texture and the way they reflect light. The frets are visible below the strings, and the background is a blurred view of the guitar's body and soundhole.

STRINGSENSE (GUITAR TUNER APP)

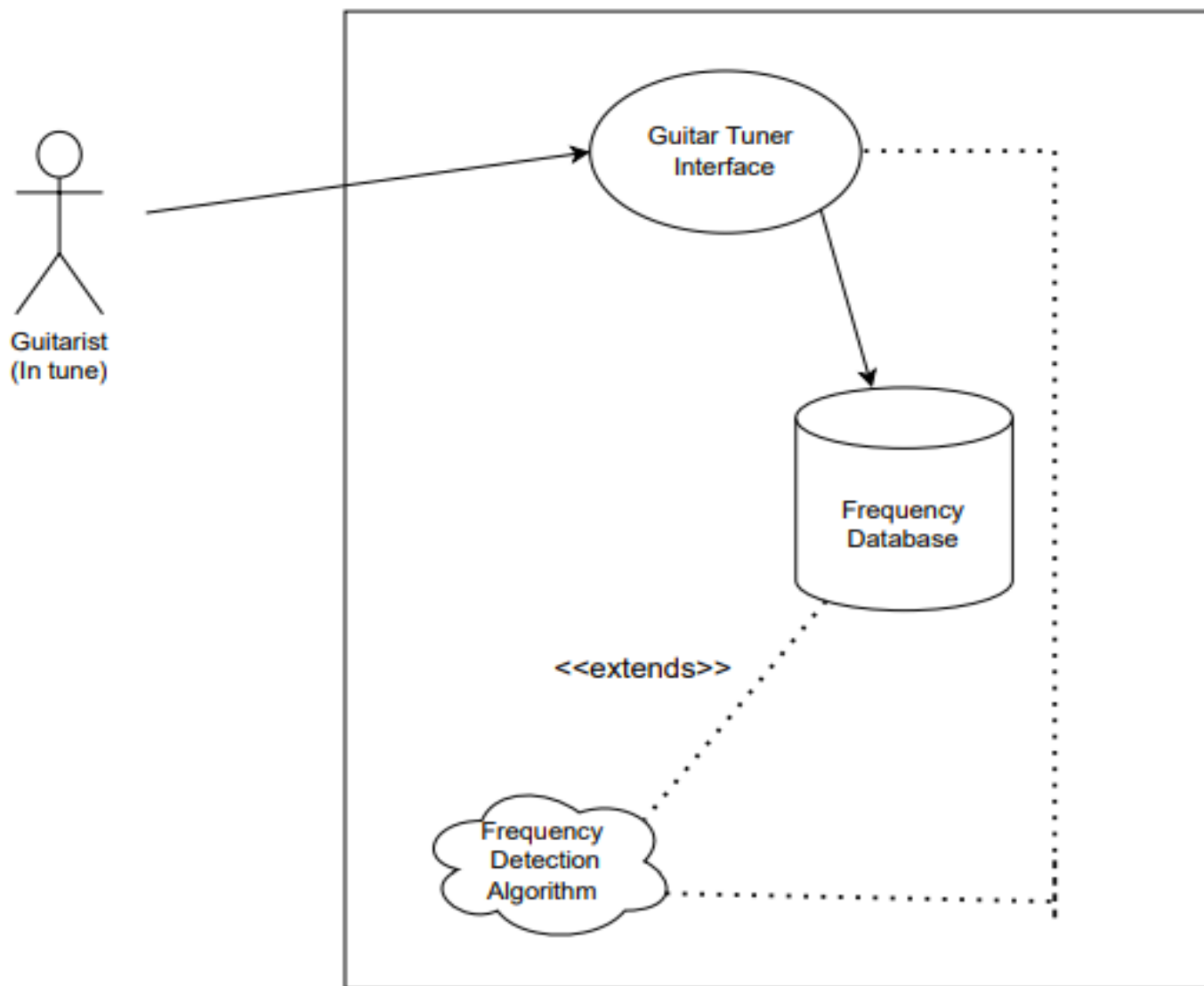
Team Rust:
Carlton Brown
Jackson Dukes
Tony Imbesi
Abri Witchett



UML SEQUENCE DIAGRAM



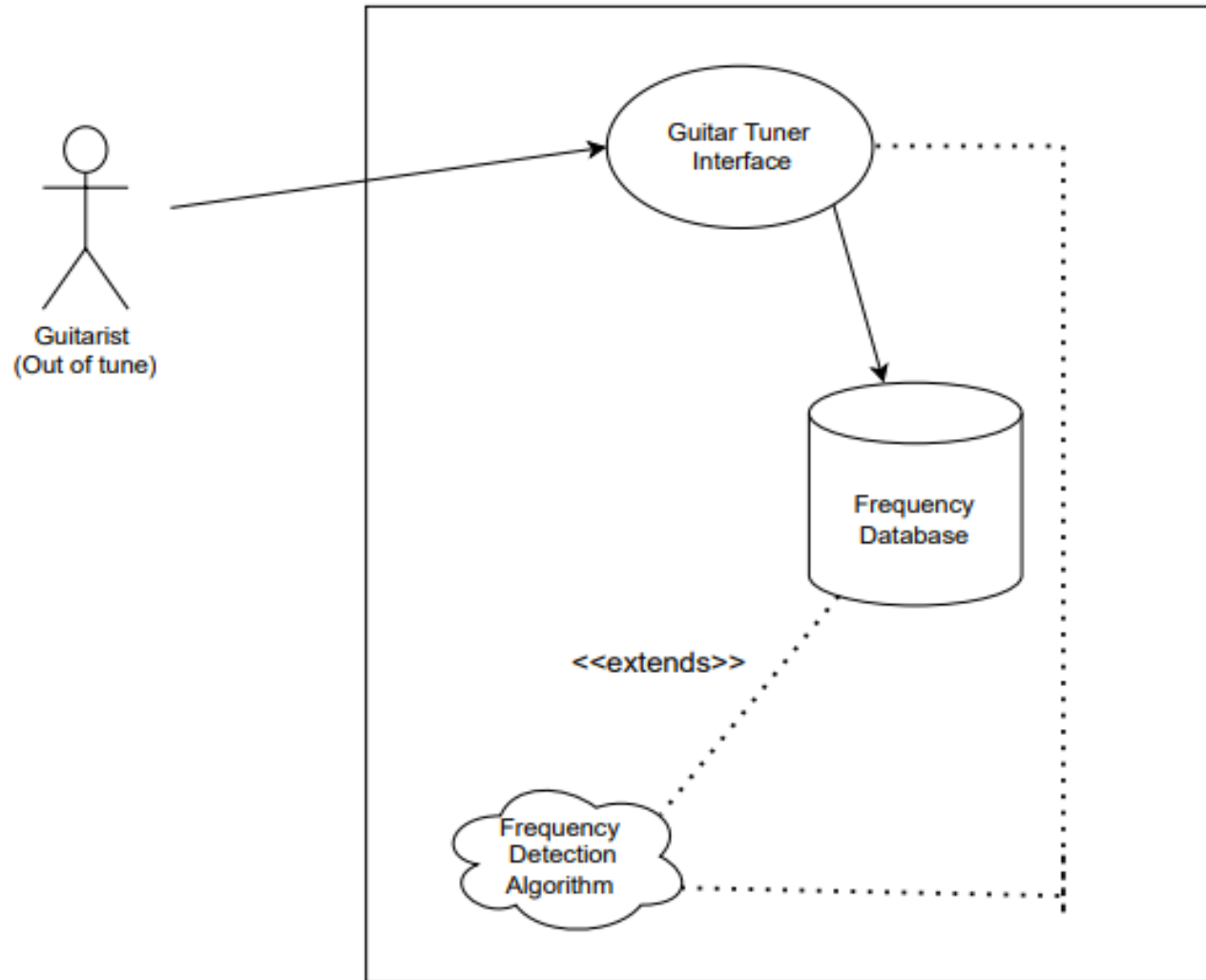
Use case Diagram 1



USE CASE SCENARIO 1 (IN TUNE)



Use case Diagram 2



USE CASE SCENARIO 2 (OUT OF TUNE)



TEST CASE SCENARIOS

- George is a hobbyist musician who does not want to buy a dedicated guitar tuner. He uses **StringsSense** instead to tune his guitar. Upon opening the app, he has the option to start tuning by pressing a button. This activates his phone's built-in microphone to start recording the sound of him plucking a guitar string.
- When George plays into the microphone, **StringsSense** extracts the frequency data from the audio input and consults a frequency-to-pitch table to find the note closest to the one the microphone received. When **StringsSense** finds that note, it shows George a color-coded indicator of how close he was to that note.



TEST CASE SCENARIOS

1. Check the user Interface

All buttons and graphics display properly

All buttons can be pressed

User can toggle microphone functionality

2. Receive tuning feedback from the app

Device microphone can receive input

Microphone accurately records guitar audio

Frequency-to-pitch lookup is accurate

