Overview:

Our objective is to create an ambient lighting system (Ambilight) for televisions or similar display devices, a weak light-source on the backside of the screen or monitor that illuminates the wall or surface behind and just around the display. Primarily the system will have three modes:

1. Video mode: Color of the lighting will extend the edges of the screen.
2. Music mode: Color of the lighting will be derived from the fast-fourier transform of the music being played.
3. Ambience mode: Lighting will not depend on any form of input and cycle through colors gradually.

Team Members:

1. Andrick Adhikari : Computer Science masters student with an undergraduate degree in Computer Science and Engineering Degree. Previous work experience includes working as a software engineer (2017-2018). Skilled programmer with ability to adapt and innovate.

2. Daniel Parada : Graduate student in the Computer Science department since 2017 (Data Science then Computer Science). Previous engineering degree and experience with acoustics theory allows me to better understand and apply FFT for the audio mode. Also, bit of a marketing genius.

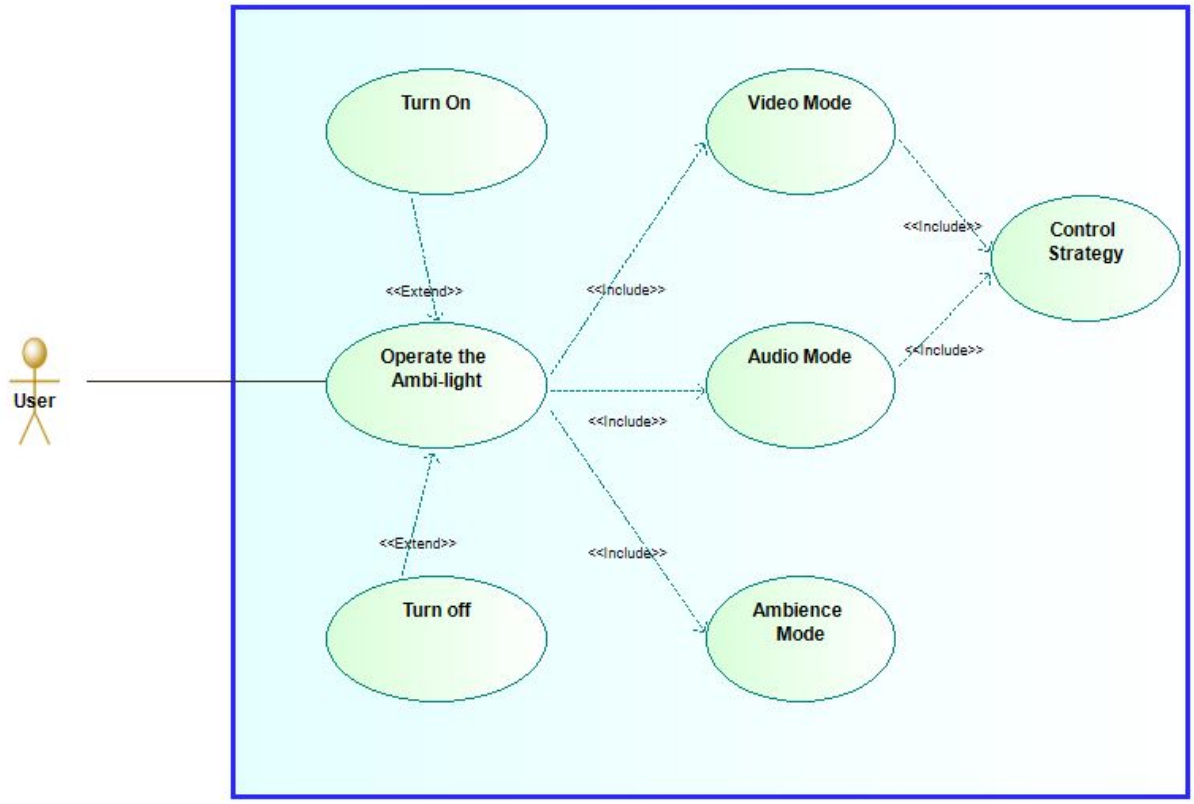
Use-Case Diagram:
Once powered on the Ambilight system will have 3 main modes of operations available to the user :

-Ambience mode : In this mode the LED lights will cycle through the available colors with a fade in and out effect.

-Audio mode : In this mode the sound sensor equipped photon board will analyze a continuous stream of audio. Based on the input, the photon board will change the color and intensity of the LEDs.

-Video mode : In this mode the system receives image data from a screen, which is preprocessed in a computer and sent to the photon board over wifi. The photon board, then generates the commands for the LED strip to match the colors of the image on the screen.

The last two modes make use of a Control Strategy. This control strategy encompasses the cycle of data input, preprocessing, transmission and output which controls the LED color and intensity display.

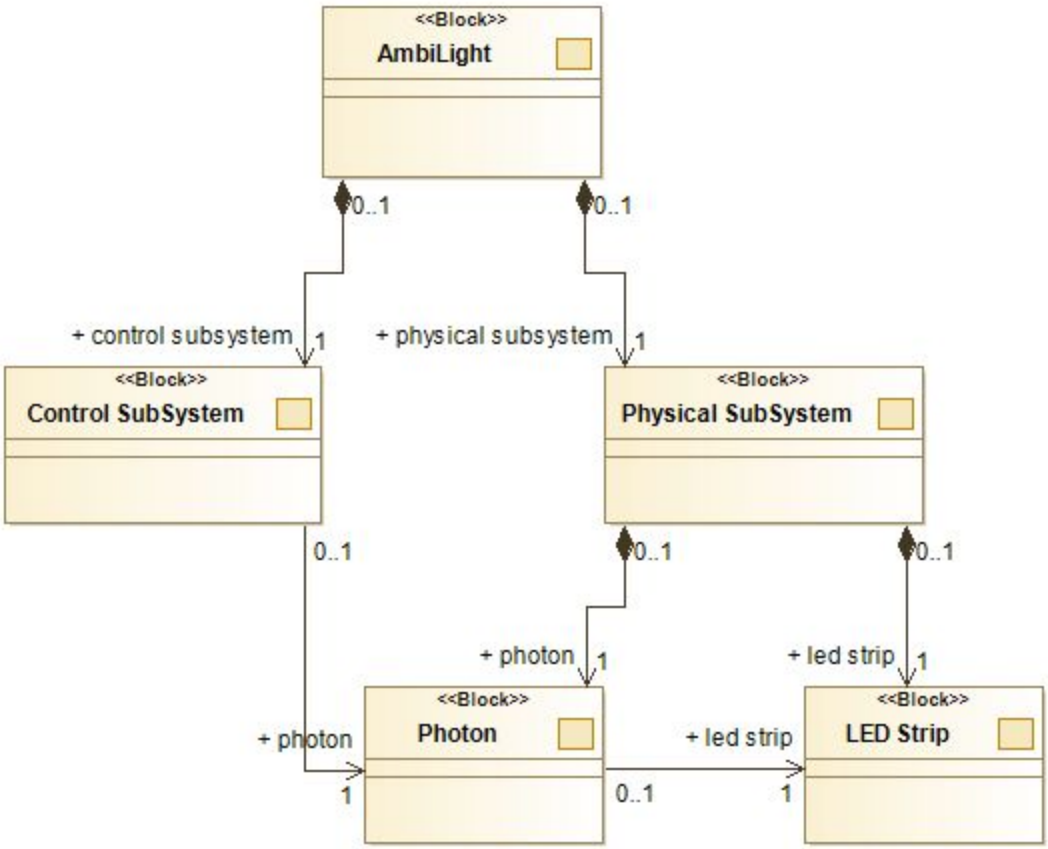


Block Definition Diagram:

The Ambilight system is composed of two main subsystems :

-Control SubSystem : This SubSystem encompasses the cycle of the input data in the system. This includes the input data stream, preprocessing (when applicable, i.e. audio and video modes), transmission of preprocessed data, generation of the LED light command.

-Physical SubSystem : This SubSystem encompasses every physical component of the Ambilight system. From the input element and power device of the entire system, through the preprocessing elements and finally the output to the LED strip.



Internal Block Diagram (PhysicalSubSystem):

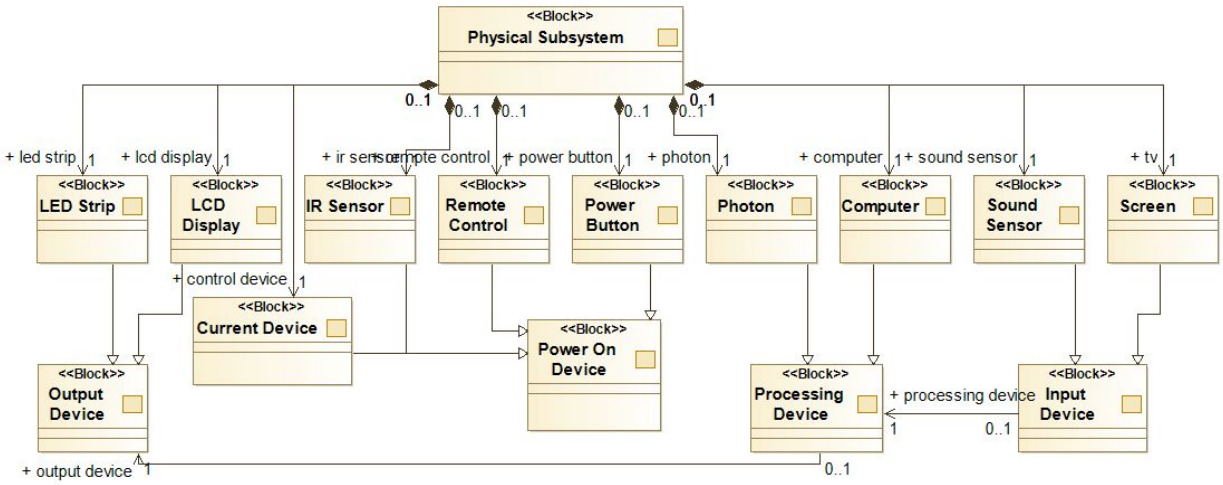
A closer look at the physical components for the Ambilight system, the elements can be split into 4 main categories :

-Power on devices : these elements are responsible for supplying the system with power as well as sending commands to start the system. Once the system is powered on, we can select a mode of operation among the 3 available ones.

-Input devices : based on the mode of operation, the input devices will change from the sound sensor on the photon board when in audio mode, to a screen when in video mode to nothing when in ambiance mode

-Preprocessing devices : when operating in audio mode, the preprocessing of data is done directly on the Particle Photon board. In video mode, given the more complex nature of the data as well as the quantity of data, preprocessing is done on a computer.

-Output devices : these are the components that will be changed in the physical space due to the processing of the data coming from our input devices. LED strip will change colors to match the sound or video input and the LCD display will display the IP address for the Particle Photon board and the mode in which the system is operating under.

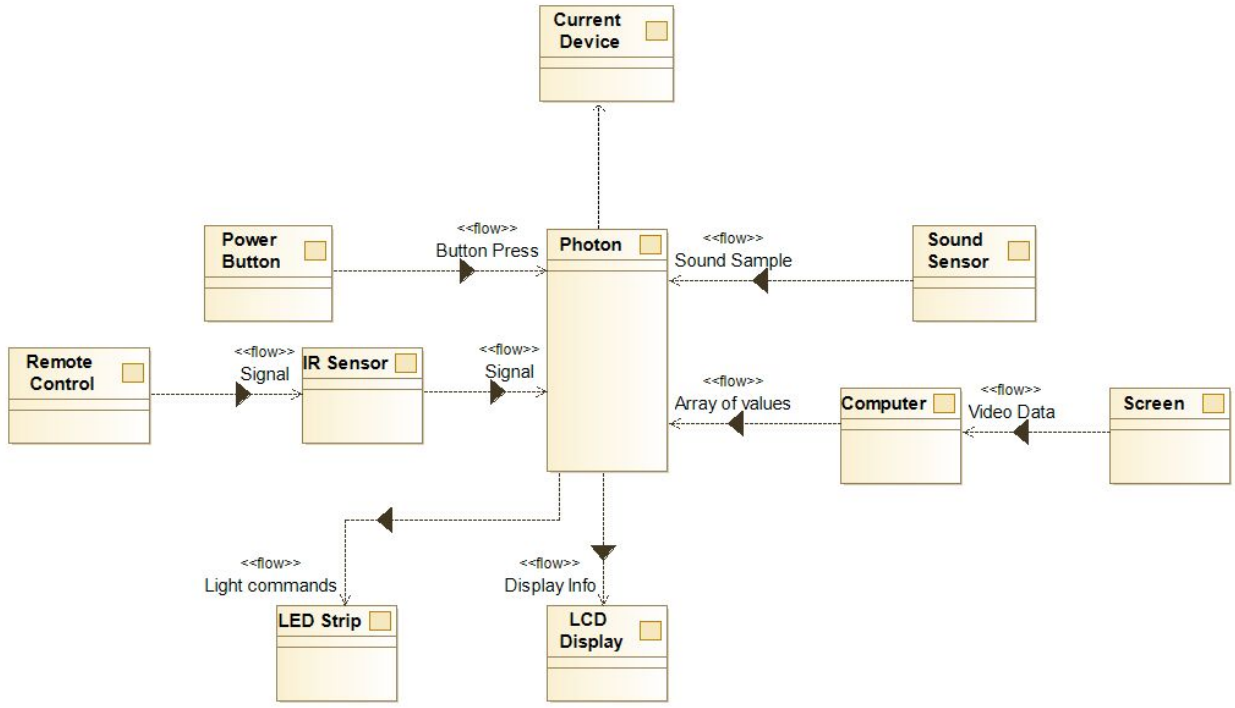


Internal Block Diagram (ControlSubSystem):
Taking a closer look at the ControlSubSystem the entire structure is centered around the Particle Photon board. The Photon board is powered by the current device and is turned on by either a touch sensitive power button or with a remote control and IR sensor.

Based on the mode of operation, the input will come from the sound sensor (sound mode) or via WiFi from the computer, which itself receives continuous input from the screen. The LCD display shows the mode in which the Photon board is operating in as well as the IP address of the board that the user must connect to when in video mode.

Sound Mode : the input from the sound sensor goes straight into the Particle Photon and analyzed on-device using an FFT to transform the signal from the temporal domain to the frequency domain. This transformation then allows the color changes of the LED lights based on the frequencies of the audio.

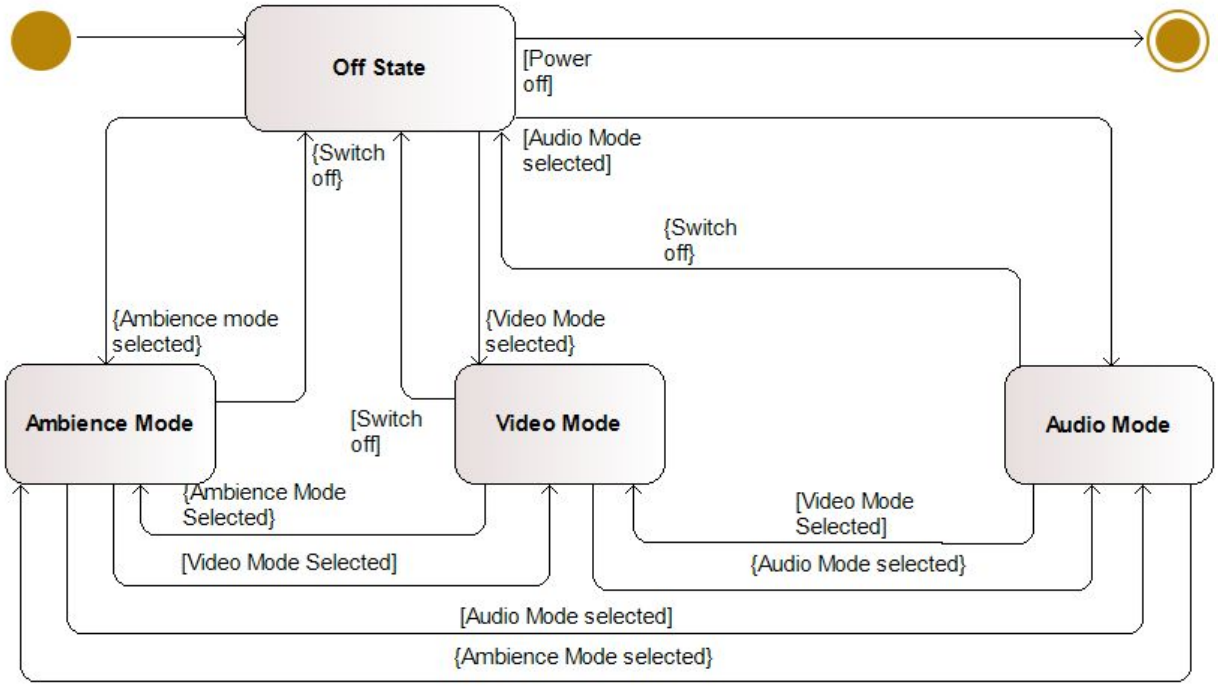
Video Mode : The input is a continuous stream of video feed which is collected by a computer for preprocessing. The input data is subdivided into regions corresponding to the regions where the LED lights are on the screen and the color per region is averaged and sent to the Photon board. The Photon board then creates the light commands which are sent to each individual LED light.



State Machine diagram :

High-level behaviour of the software is represented by the state diagram below. The system will start in the "**off state**", where it will be idle and wait for signal from the IR receiver and based on the mode selected switch to either "**ambience mode", "video mode" or "audio mode".** Also, user can go back to any of the other modes or off state through appropriate remote control input.

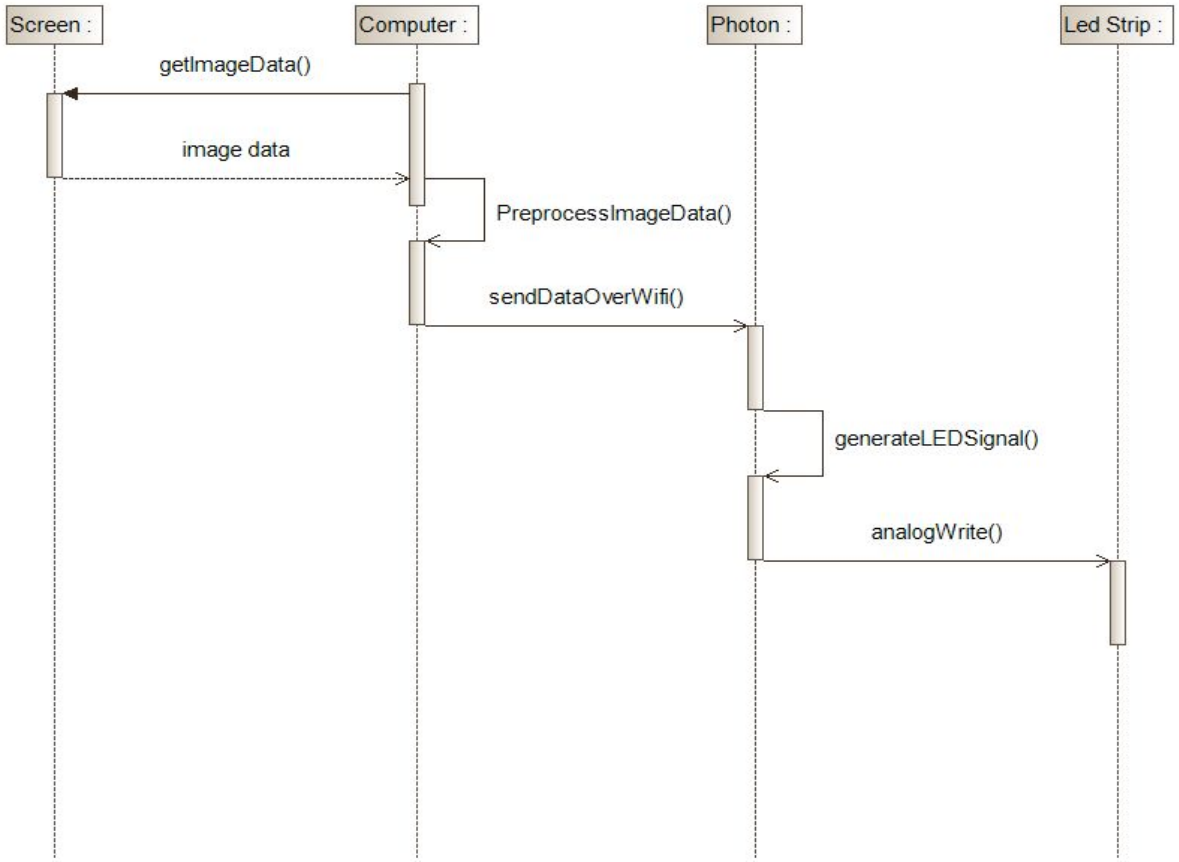
Further description of each of the three modes is explained below with respective sequential diagrams.



Video Mode:

It is going to operate by continuously executing following methods in the loop:

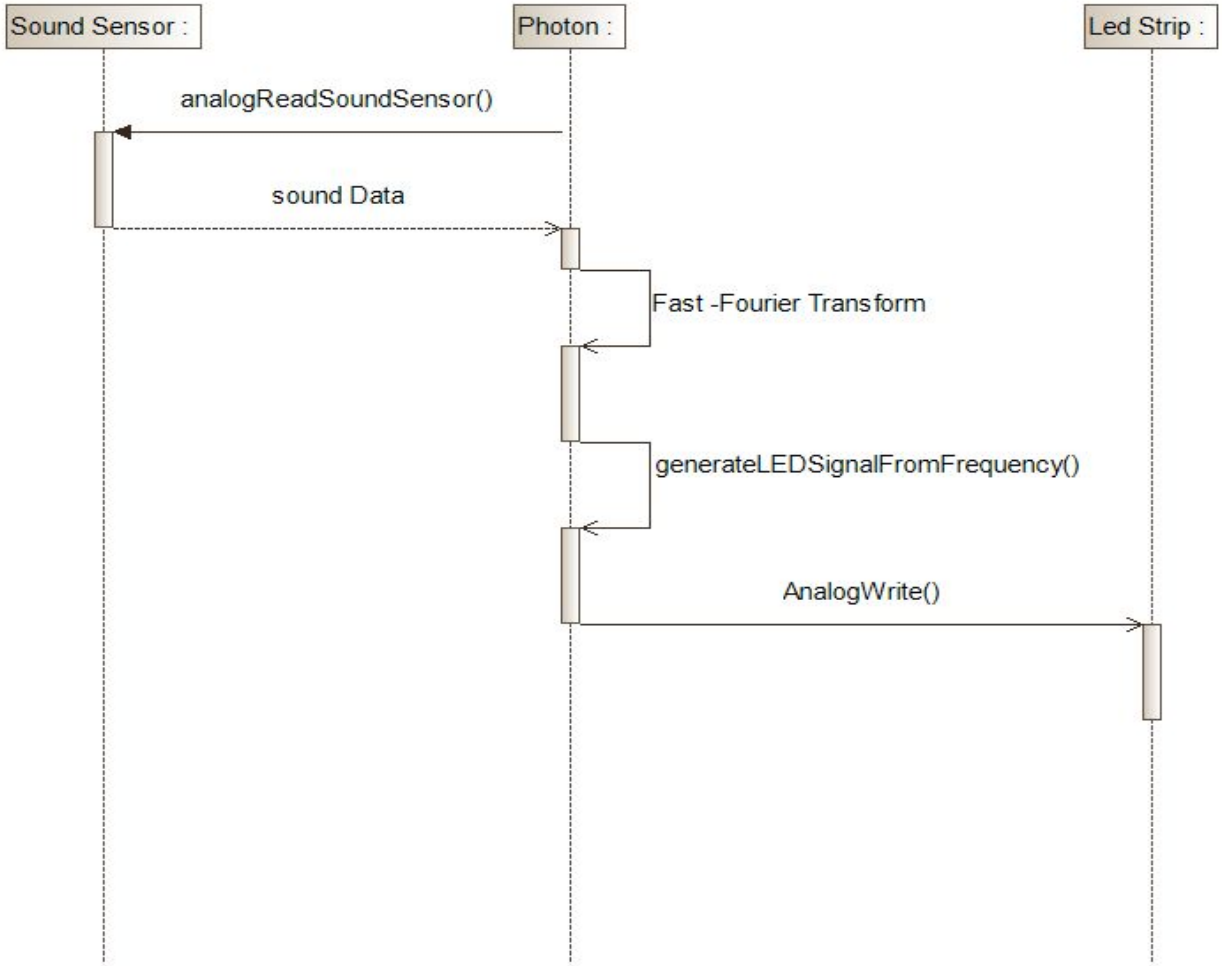
a. getImageData(): Computer is gonna read the display information (pixel values) from the screen.
b. preProcessImageData(): Computer is going to process the pixel values of the screen to compute the required edge light values and intensity.
c. sendDataOverWifi(): Computer is going to send data to Photon over wifi.
d. generateLEDSignal(): Photon is going to process the received data from computer and decide on appropriate values for each led. Also, it is going to make sure that led color values don't change drastically which can be eye straining for the viewer.
e. analogWrite(): Photon is going to write the values to LED Strip.



Audio Mode:

Similar to Video Mode, audio mode is going to execute the following methods in a loop:

a. analogReadSoundSensor(): Photon is going to read the input from sound sensor.
b. Fast-Fourier Transform: Photon will change sound data from temporal to frequency space.
c. generateLEDSignalFrequency(): Based on the sound data in frequency space, this method will compute values for each of the LED.
d. AnalogWrite(): Photon is going to write the values to LED Strip.
e.



Sequence Diagram : Ambiance mode

Light colors cycle through all available RGB colors over time.