Descripcion del Proyecto

## Propósito, Objetivos y Alcance del Proyecto

El propósito es desarrollar un compilador para un lenguaje gráfico que permita a los usuarios programar música mediante la selección la utilización de estructuras de control y elementos musicales, todo con el fin de aprender a programar de una manera más interactiva y dinámica mediante la exploración de composiciones musicales.

El objetivo del lenguaje es enseñar programación a niños de secundaria mediante la utilización de bloques gráficos para que puedan entender estructuras de control, lógica de programación y que al mismo tiempo exploten su lado creativo que sería la producción de obras musicales o imitación de las mismas mediante estructuras de control.

El alcance está definido en que utilizará estructuras básicas de control (if’s, for’s, while’s) junto con operaciones básicas y el uso de módulos o funciones para poder programar en bloques gráficos sencillos para finalmente bajar el código y ejecutar para sacar un archivo .wav que representa los plays que ingresó el usuario en el orden en que los programó y se ejecutaron de acuerdo a la visión lógica del mismo. Es un lenguaje con operaciones básicas que permiten analizar un diseño lógico sencillo mediante la salida de notas para poder componer melodías de un solo instrumento.

## Análisis de Requerimientos y Casos de Uso generales

El requerimiento principal es que sea un lenguaje que pueda ser escrito mediante una interfaz gráfica y que a través de la compilación se puedan respetar los estatutos básicos de un lenguaje y al mismo tiempo produzca música como salida. Por lo que el requerimiento es que a través de una interfaz gráfica pasemos a una sintaxis particular en un archivo de texto que el usuario puede escribir directamente si quiere esa libertad, para finalmente compilar todo esto utilizando python como maquina virtual que lee y ejecuta los cuadruplos generados al parsear el lenguaje.

Los casos de uso son los definidos en el diagrama:



Aquí podemos ver que la compilación puede ser exitosa mediante el flujo de blockly -> archivo.txt -> terminal para finalmente ver lo que se despliega en consola y el archivo .wav que se genera si es que se incluyen plays. El flujo alterno en una compilación exitosa es brincarse blockly e ir directo a escribir en nuestro lenguaje al archivo.txt -> terminal para que genere los outputs en consola y el .wav según sea el caso. Todos los errores de compilación existirán hasta el momento en el que se utilice la consola, blockly solo está para que la programación sea más dinámica y sencilla. Los errores pueden ir desde léxico (palabras que no acepta), sintaxis (reglas específicas de escribir), semántica (acciones que corresponden a ciertos tipos de datos), hasta ejecución (divisiones entre 0, índices fuera de rango etc.).