

**Calculadora Científica basada en Win32 API**

**Especificación de requerimientos**

**Integrantes:**

* *Ricardo Alejandro Grimaldo Patiño*
* *Eduardo Zenet López Guerrero*
* *Edgar Bezares Samaniego*

**Licenciatura en Ingeniería de Software**

Segundo Semestre

**Programación estructurada**

Profesor: *M.I.T Edwin Jesús León Bojorquez*

*Domingo 14 de junio del 2020; Mérida, Yucatán*

**Índice**

[1. Introducción 3](#_Toc43059342)

[2. Propósito 3](#_Toc43059343)

[3. Objetivo 3](#_Toc43059344)

[4. Funcionalidades del producto 4](#_Toc43059345)

[5. Interfaz gráfica de usuario 4](#_Toc43059346)

[6. Requerimientos funcionales 4](#_Toc43059347)

[7. Requerimientos no funcionales 6](#_Toc43059348)

[8. Requerimientos de interfaz gráfica 8](#_Toc43059349)

[9. Casos de uso 9](#_Toc43059350)

[Funciones de conversión 9](#_Toc43059351)

[Funciones trigonométricas 12](#_Toc43059352)

[Jerarquía de operaciones 15](#_Toc43059353)

[10. Plan de Trabajo 18](#_Toc43059354)

[Herramientas 18](#_Toc43059355)

[● Aplicación de Code::Blocks 18](#_Toc43059356)

[● Google Drive 18](#_Toc43059357)

[● WhatsApp 18](#_Toc43059358)

[● Git/Github 19](#_Toc43059359)

[● Windows API 19](#_Toc43059360)

[Roles 20](#_Toc43059361)

# 

# 1. Introducción

El presente documento ha sido elaborado bajo el estándar IEEE 830-1998 con el fin de la especificación de requerimientos de software, de modo que contiene los requerimientos ligados al producto software ***“Calculadora Científica basada en Win32 API”***, tanto funcionales como no funcionales, de modo que se especificarán las características del programa relacionadas a su uso, operamiento y función, así como con la portabilidad y manejo del mismo.

De la misma manera, se incluyen los requisitos necesarios sobre la interfaz gráfica, así como los diferentes casos de uso posibles para el uso del software y las herramientas involucradas en el proceso de desarrollo del mismo.

# 2. Propósito

Este programa surge a raíz de la dificultad hacia algunas personas para resolver problemas matemáticos donde estén presentes elementos de jerarquía de operaciones, funciones trigonométricas y funciones de conversión, así como otras operaciones disponibles, incluidas en el producto final.

# 3. Objetivo

Una vez presentada la problemática, la Calculadora Científica basada en Win32 API tiene como objetivo resolver dichas operaciones, como evaluación de una expresión matemática mediante la jerarquía de operaciones y resolución con funciones trigonométricas, además de tener la capacidad de convertir un número decimal a su equivalente en grados, minutos y segundos, y finalmente, convertir un número entero a su misma expresión en números hexadecimales, binarios y octales, todo por medio de una agradable a la vista y de fácil uso para la comodidad de nuestros usuarios.

De este modo, el presente documento tiene el propósito de especificar los tipos de requerimientos relacionados al producto software, a fin de documentar las funciones que realizará en este programa de acuerdo a la funcionalidad asociada, describiendo su operatividad y para qué sirve, especificando y aclarando la manera en la que trabajará el mismo y cómo se comportará.

# 4. Funcionalidades del producto

* Jerarquía de operaciones
* Operaciones algebraicas básicas
* Operaciones algebraicas adicionales
* Conversión de un número entero a su equivalente en binario, decimal y hexadecimal
* Cálculo de funciones trigonométricas, sus inversas y recíprocas.
* Alertas de diversos tipos de errores
* Manejo de interfaz gráfica

# 5. Interfaz gráfica de usuario

Para una interacción con el usuario y mostrar de manera tanto visual como gráfica al mismo, se llevó a cabo el uso y apoyo de bibliotecas y funciones de Windows API, capaces de mostrar una interfaz gráfica con entrada y salida de datos, todo programado desde el lenguaje C.

En el caso de nuestro proyecto, se presentará una calculadora científica con diversas funciones y operaciones matemáticas (todas presentadas en la sección de **requerimientos funcionales**) de manera visual, con la cual el usuario será capaz de interactuar por medio del mouse (botones).

# 6. Requerimientos funcionales

**RF 1:** *Entrada*. El programa leerá una entrada de una línea de operaciones aritméticas a la vez desde una caja de texto y calculará el resultado final.

**RF 2:** *Jerarquía*. El programa leerá la entrada ingresada considerando la jerarquía matemática estándar de operadores:

1. Primero se realizarán todas las operaciones que estén dentro de paréntesis.
2. Luego se realizarán las racionalizaciones y funciones trigonométricas, así como sus inversas y recíprocas.
3. Posteriormente, las potenciaciones,factoriales y porcentajes.
4. Después se realizarán las multiplicaciones y divisiones.
5. Finalmente se harán las sumas y restas.

**RF 8:** *Operaciones algebraicas básicas.* La calculadora realizará las operaciones básicas tales como suma, resta, multiplicación y división de 2 números introducidos, así como su porcentaje.

**RF 9:** *Racionalización*. El programa será capaz de realizar la raíz cuadrada de un valor si el usuario hace click en el botón ‘sqrt(x)’, donde ‘x’ es el número al que se le planea sacar la raíz, o bien, escribiendo desde el teclado dicha función de manera correcta.

**RF 10:** *Potencia.* El programa podrá calcular la potencia de un número, siempre y cuando el usuario ingrese la sintaxis correcta para éste, haciendo click en el botón con el símbolo ‘^’, donde el valor situado antes del mismo es el número base y el valor situado después de éste es el número de veces que se desea elevar dicho número base.

**RF 11:** *Porcentaje*. El programa podrá realizar la operación ‘Porcentaje’. La operación ‘Porcentaje’ de un número se define como el cálculo de la división entre un número y 100. Es decir, si tenemos un número ‘x’, la expresión ‘x%’ será equivalente a la operación ‘x/100’. Esta función se ejecutará, ya sea al leer como entrada el carácter ‘%’, o pulsando el botón con el mismo símbolo en la interfaz, justo después del valor al que se le desea realizar la operación.

**RF 12:** *Factorial*. El programa también considerará el cálculo de la operación ‘Factorial’. La operación ‘Factorial’ de un número se define como el cálculo de la multiplicación de todos los números anteriores a éste, teniendo como mínimo número a consideración el 1. Esta función se ejecutará, ya sea al leer como entrada el carácter ‘!’, o pulsando el botón con el mismo símbolo en la interfaz, justo después del valor al que se le desea sacar su factorial.

**RF 13:** *Funciones trigonométricas*. El programa podrá leer y realizar las funciones trigonométricas ‘seno’ (sin), ‘coseno’ (cos), ‘tangente’ (tan), así como sus recíprocas, que son ‘cotangente’ (cot), ‘secante’ (sec) y ‘cosecante’ (csc), siendo el usuario capaz de usarlas por medio de los botones con sus respectivos nombres abreviados, o bien, escribirlas desde el teclado (las funciones trigonométricas se calcularán en radianes).

**RF 14:** *Funciones inversas*. El programa podrá leer y realizar las funciones trigonométricas inversas ‘arcoseno’ (arcsin), ‘arcocoseno’ (arccos), y ‘arcotangente’ (arctan), siendo el usuario capaz de usarlas por medio de los botones con sus respectivos nombres, o en su defecto, escribiéndolas correctamente desde el teclado (las funciones inversas se calcularán en radianes).

**RF 15:** *Conversión a grados*. El programa podrá transcribir el resultado de una operación previa (siempre y cuando sea positivo) a la forma de grados, minutos y segundos (x° y’ z’’), donde ‘x’ es el número dedicado para los grados, ‘y’ es el número dedicado para los minutos y ‘z’ es el número dedicado para los segundos.

**RF 16:** *Conversión a binario*. El programa podrá convertir el valor del resultado de una operación previa al sistema de numeración binario. El sistema de numeración binario consta únicamente de dos dígitos, los cuales son el ‘0’ y el ‘1’.

**RF 17:** *Conversión a octal*. El programa podrá convertir el valor del resultado de una operación previa al sistema de numeración octal. El sistema de numeración octal consta de 8 dígitos: ‘0’, ‘1’, ‘2’, ‘3’, ‘4’, ‘5’, ‘6’ y ‘7’.

**RF 18:** *Conversión a hexadecimal*. El programa podrá convertir el valor del resultado de una operación previa al sistema de numeración hexadecimal. El sistema de numeración hexadecimal consta de 16 dígitos: ‘0’, ‘1’, ‘2’, ‘3’, ‘4’, ‘5’, ‘6’, ‘7’, ‘8’, ‘9’, ‘A’, ‘B’, ‘C’, ‘D’, ‘E’ y ‘F’.

# 7. Requerimientos no funcionales

**RNF 1:** *Alertas*. El programa mandará un alerta de ***‘Error’*** al detectar uno de los 4 posibles tipos de errores, los cuales pueden ser: Sintáctico, matemático, léxico o de límite.

**RNF 2:** *Error léxico*. El sistema detectará cuando en la entrada se intente ingresar algún carácter no válido. Un carácter se considerará no válido cuando este no se pueda ingresar desde los botones de la interfaz gráfica. Cuando se encuentre un carácter no válido, al momento de querer calcular la expresión, el sistema mandará un alerta al usuario informando acerca de dicho error.

**RNF 3:** *Error sintáctico*. El sistema detectará cuando se use erróneamente la sintaxis de las distintas operaciones disponibles, por ejemplo, cuando el usuario intente ingresar “2++2”. Cuando se encuentre un manejo incorrecto de la sintaxis, al momento de querer calcular la expresión, el sistema mandará un alerta al usuario informando acerca de dicho error.

**RNF 4:** *Error matemático*. El sistema detectará cuando en la entrada aparezca una operación indefinida, ya sea por el límite de la función (por ejemplo, arccos(6)), o por la operación en sí (por ejemplo, una división entre cero). Cuando el resultado de una operación sea indefinida o imposible de calcular, el sistema mandará un alerta al usuario informando acerca de dicho error.

**RNF 5:** *Error de límite*. El sistema detectará cuando el resultado de una operación válida sobrepase el límite máximo de impresión de caracteres (30). Cuando el resultado a imprimir sobrepase el límite, el sistema mandará un alerta al usuario informando acerca de dicho error.

**RNF 6:** *Lectura*. El programa sólo leerá y realizará una línea de entrada por vez, y la lectura de la línea para el cálculo de operaciones se llevará a cabo de izquierda a derecha.

**RNF 7:** *Acceso*. El acceso al programa será limitado para un solo usuario a la vez.

**RNF 8:** *Seguridad*. No se solicitará ningún tipo de registro o autenticación para acceder al programa.

**RNF 9:** *Límites de la interfaz*. La interfaz gráfica programada con Win32 API estará delimitada por las siguientes secciones: Área de ingreso de entradas y de impresión de salidas, Área de teclado numérico, Área de conversiones y Área de funciones.

**RNF 10:** *Eficiencia*. Cada función deberá responder al usuario en un lapso máximo de 2 segundos.

**RNF 11:** *Manejo de funciones de conversión*. Las funciones ‘Conversión a grados’ (°), ‘Conversión a binario’ (BIN), ‘Conversión a octal’ (OCT), ‘Conversión a hexadecimal’ (HEX), siempre imprimirán el resultado de la conversión en el único caso de que sean números positivos, caso contrario, permanecerán en blanco.

**RNF 12:** *Parte decimal en las funciones de conversión*. Las funciones ‘Conversión a binario’ (BIN), ‘Conversión a octal’ (OCT), ‘Conversión a hexadecimal’ (HEX), sólo aceptarán la parte entera del resultado de una operación previa, omitiendo la parte decimal.

**RNF 13**: *Entrada de caracteres*. El número máximo de caracteres en la entrada del usuario será de **30**.

**RNF 14:** *Límite del resultado*. El límite de dígitos de la impresión del resultado será de **18**.

**RNF 15:** *Interfaz gráfica*. La calculadora será presentada por medio de una interfaz gráfica programada en C, utilizando el conjunto de funciones Win32 API.

**RF 16:** *Resultado*. El resultado de la operación/función se mostrará en pantalla toda vez que el usuario oprima el botón de ‘=’ presentado en la interfaz gráfica.

**RF 17:** *Limpieza*. En el caso de que el usuario decida realizar una limpieza del cuadro de texto, para borrar el resultado o la entrada anterior, el usuario deberá apretar el botón con las letras ‘AC’ presentado en la interfaz gráfica, borrando no sólo lo presentado en la caja de texto, sino que a su vez, se borrará lo que esté en las etiquetas de conversiones.

**RF 18:** *Apagado*. El usuario decidirá si quiere apagar el programa, haciendo click en el botón con las letras ‘OFF’ presentado en la interfaz gráfica o, si así lo desea, cerrar el programa con el botón de cerrar habitual de los programas en Windows.

# 8. Requerimientos de interfaz gráfica

* Sistema operativo de 32 o 64 bits.
* Windows 7 o posterior.
* Almacenamiento mínimo requerido en computadora de 80 KB

# 9. Casos de uso

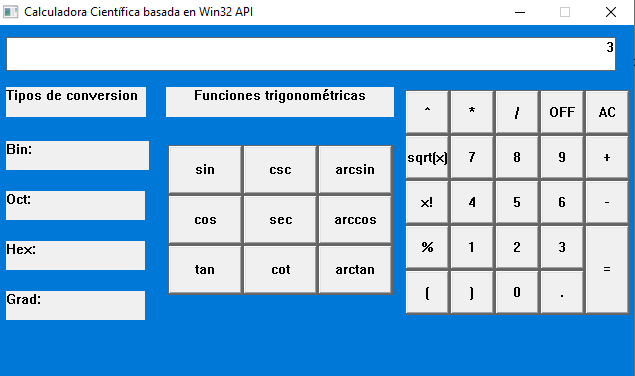
A continuación se presentarán diversos casos de uso del sistema junto a su respectiva descripción e imágenes de la pantalla de la aplicación.

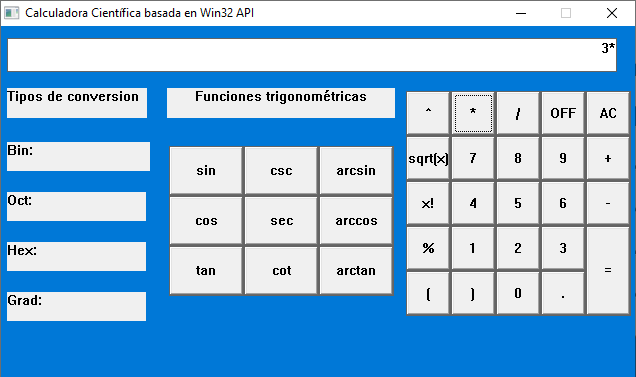
## Funciones de conversión

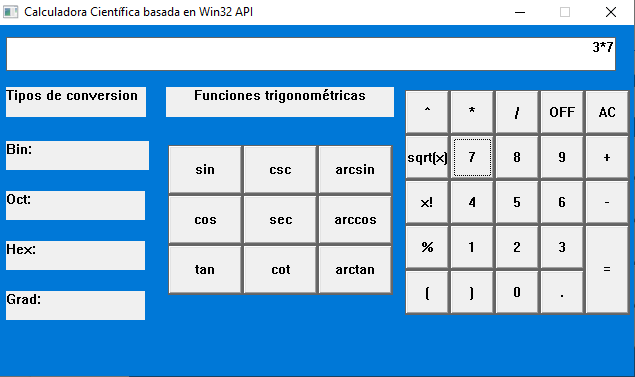
*Paso 1: Ingresar la entrada de operaciones.*

**Descripción:** Primero, el usuario deberá ingresar una operación matemática válida en la línea de texto, utilizando el teclado físico o los botones implementados en la interfaz gráfica.

**Evento del sistema:** El sistema irá agregando lo escrito o elegido por el usuario en la caja de texto presentada.



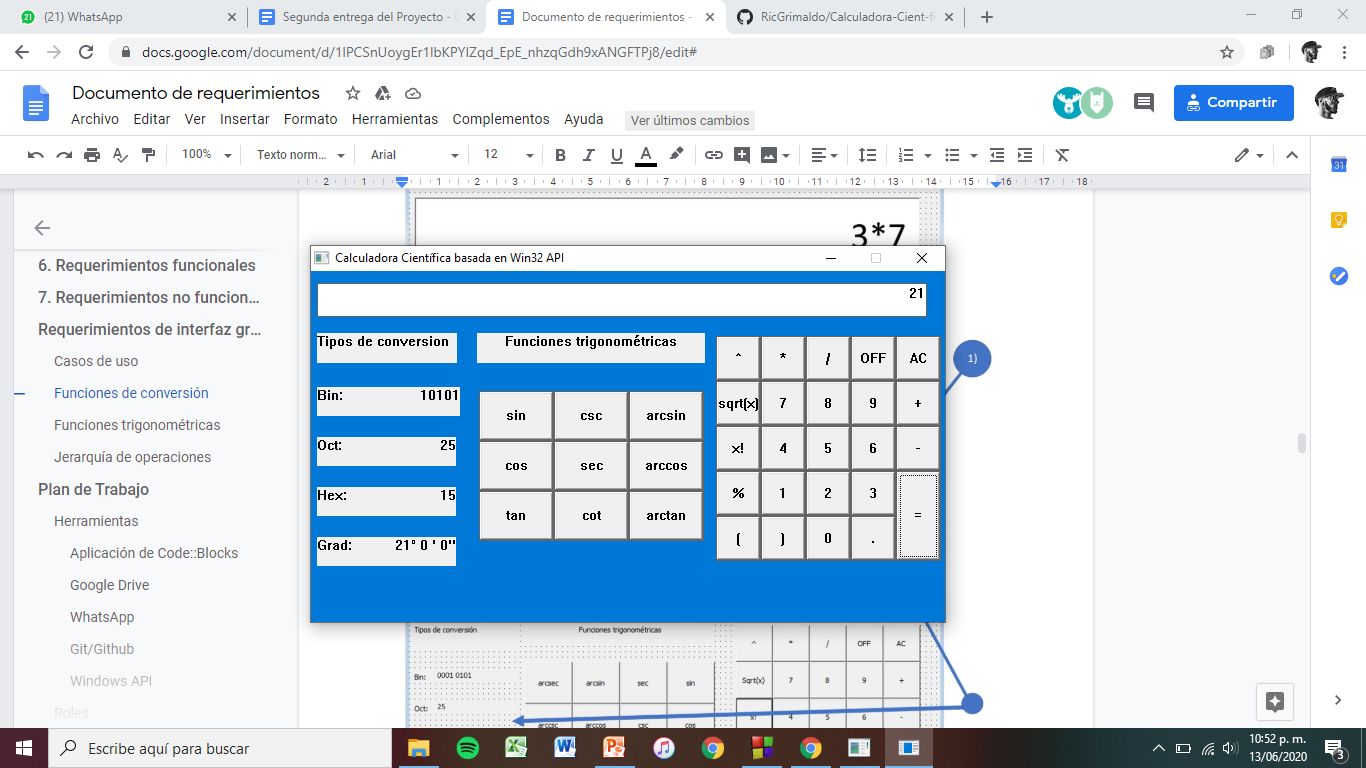




*Paso 2: Obtener resultado.*

**Descripción:** El usuario deberá hacer click en el botón “Igual a” (=) implementado en la interfaz gráfica para obtener el resultado deseado.

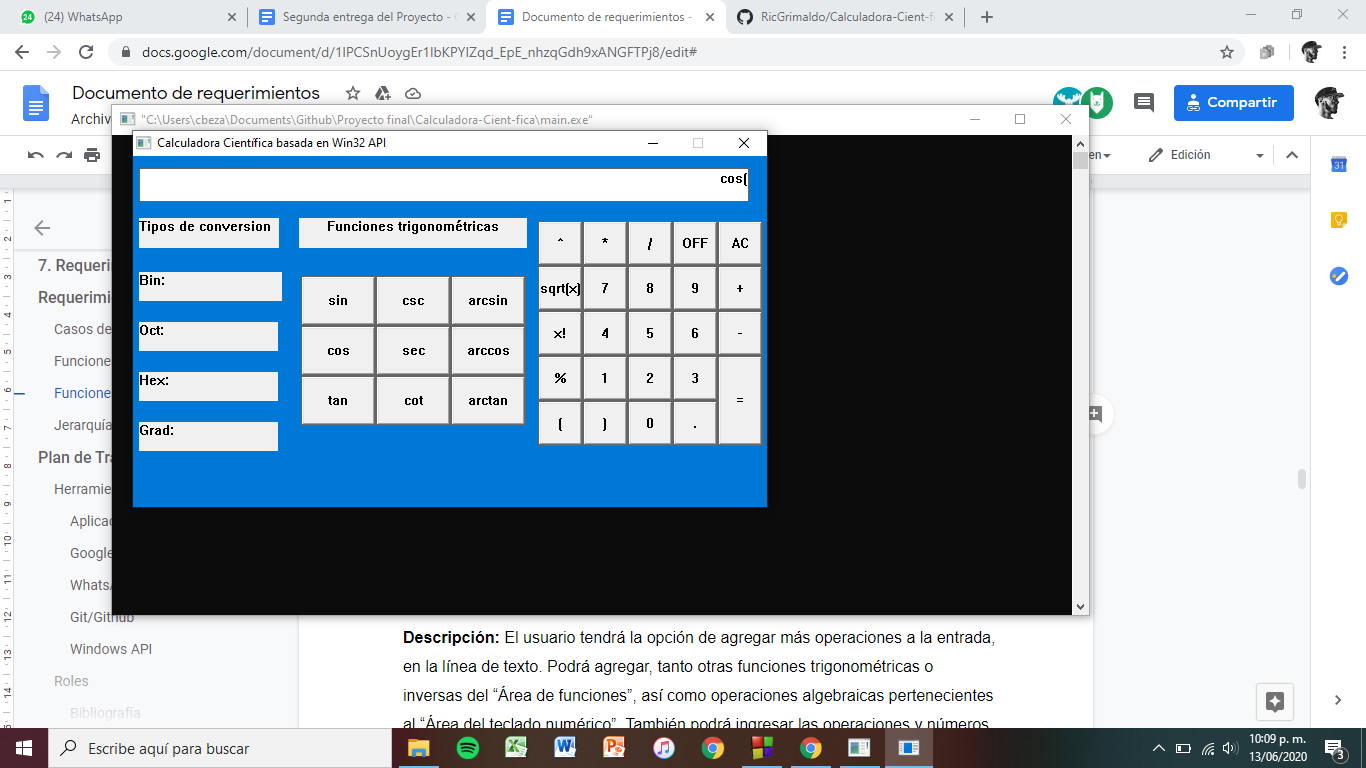
**Evento del sistema:** El sistema mostrará el resultado en la caja de texto, reemplazando la expresión anteriormente escrita, al mismo tiempo que ésta será usada por cada una de las funciones de conversión (binario, octal,hexadecimal,grados) para realizar su respectiva operación, y se imprimirán los resultados de la conversión en el “Área de conversiones” de la calculadora.



## Funciones trigonométricas

*Paso 1: Uso de funciones trigonométricas.*

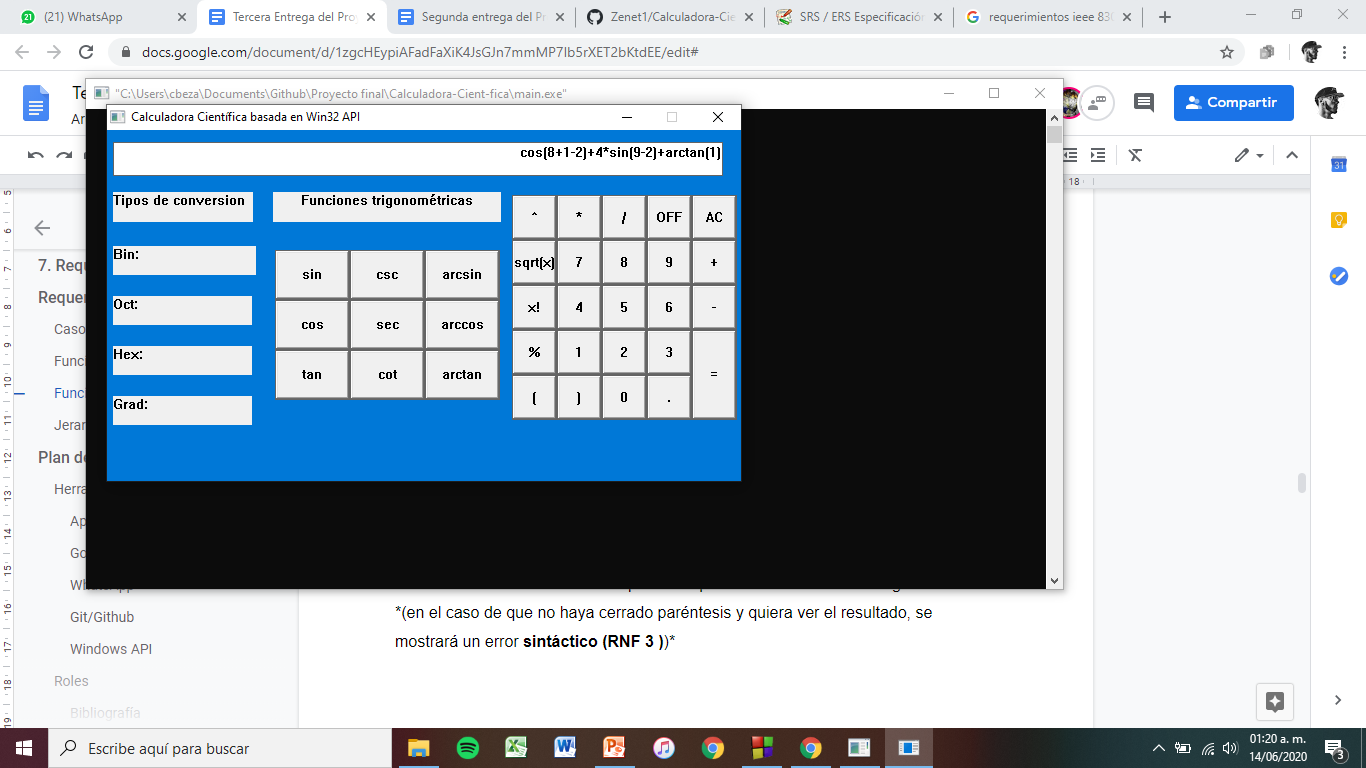
**Descripción:** Utilizar como mínimo una función trigonométrica o inversa, del “Área de funciones” de la calculadora. El usuario deberá presionar el botón implementado en la interfaz gráfica correspondiente a la función que desea realizar, o en su defecto, escribirlo desde el teclado de manera correcta.

**Evento del sistema:** El sistema mostrará en la caja de texto la función seleccionada junto a un paréntesis abierto.

*Paso 2: Opción de agregar más operaciones.*

**Descripción:** El usuario tendrá la opción de agregar más operaciones a la entrada, en la línea de texto. Podrá agregar, tanto otras funciones trigonométricas o inversas del “Área de funciones”, así como operaciones algebraicas pertenecientes al “Área del teclado numérico”. También podrá ingresar las operaciones y números requeridos desde el teclado, haciendo posible el cálculo de operaciones dentro del paréntesis y expresando el resultado en radianes.

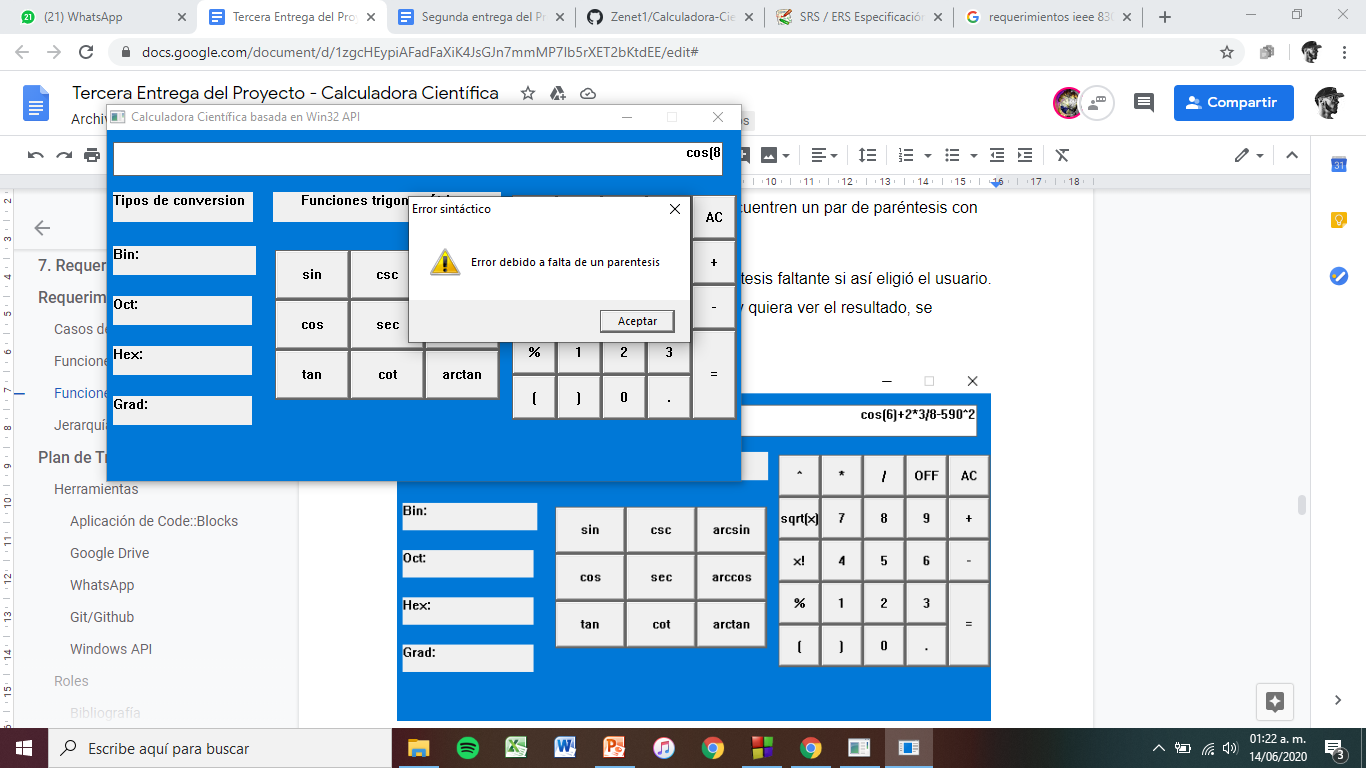
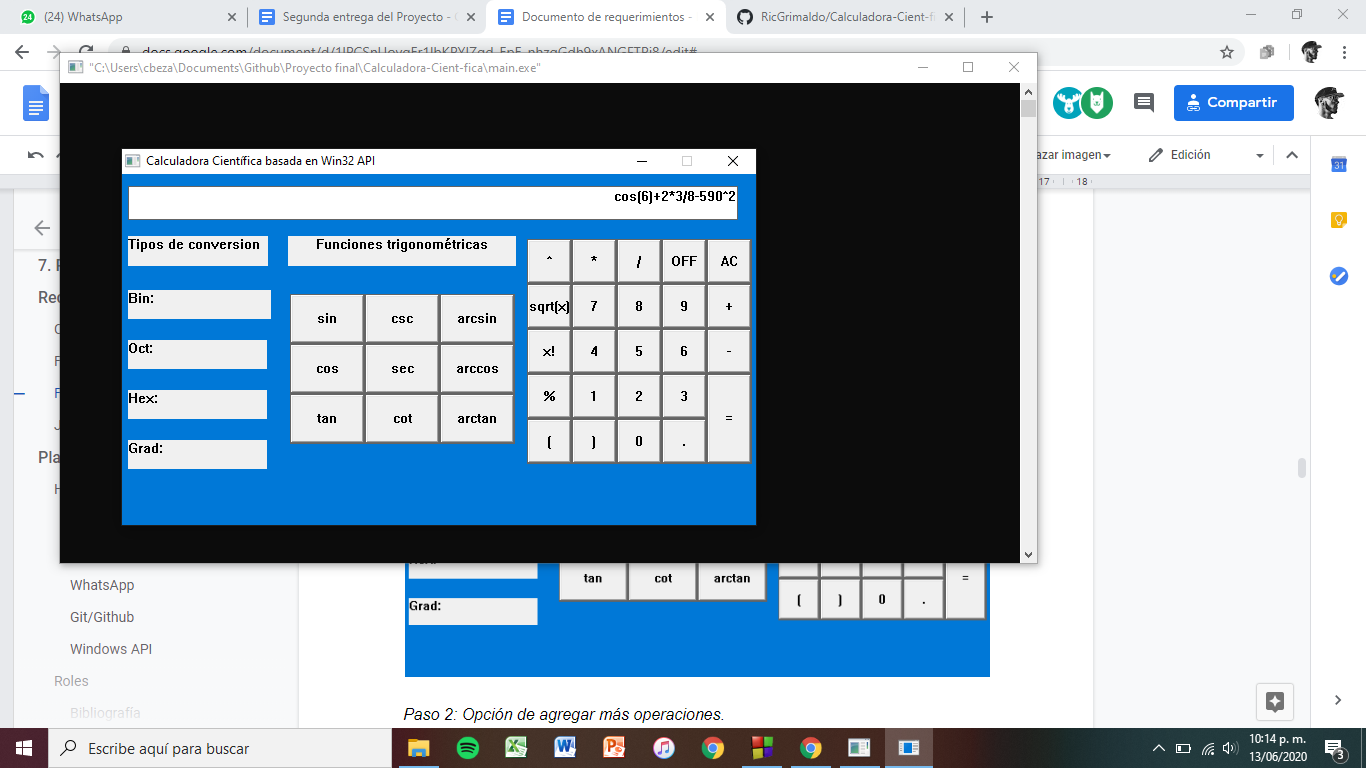
**Evento del sistema:** Todo lo ingresado por el usuario, ya sea desde el teclado o a través de los botones, se irá mostrando simultáneamente en la caja de texto.



*Paso 3: Uso de paréntesis.*

**Descripción:** El usuario deberá asegurarse de que, inmediatamente después de la o las funciones trigonométricas utilizadas, se encuentren un par de paréntesis con un número u operación de números válidos.

**Evento del sistema:** El sistema pondrá el paréntesis faltante si así eligió el usuario. \*(en el caso de que no haya cerrado paréntesis y quiera ver el resultado, se mostrará un error **sintáctico [RNF 3]**)\*



*Paso 4: Obtener resultado.*

**Descripción:** El usuario deberá desplegar el resultado de la operación ingresada previamente, presionando el botón *“Igual a”* (=) en el botón implementado en la interfaz gráfica.

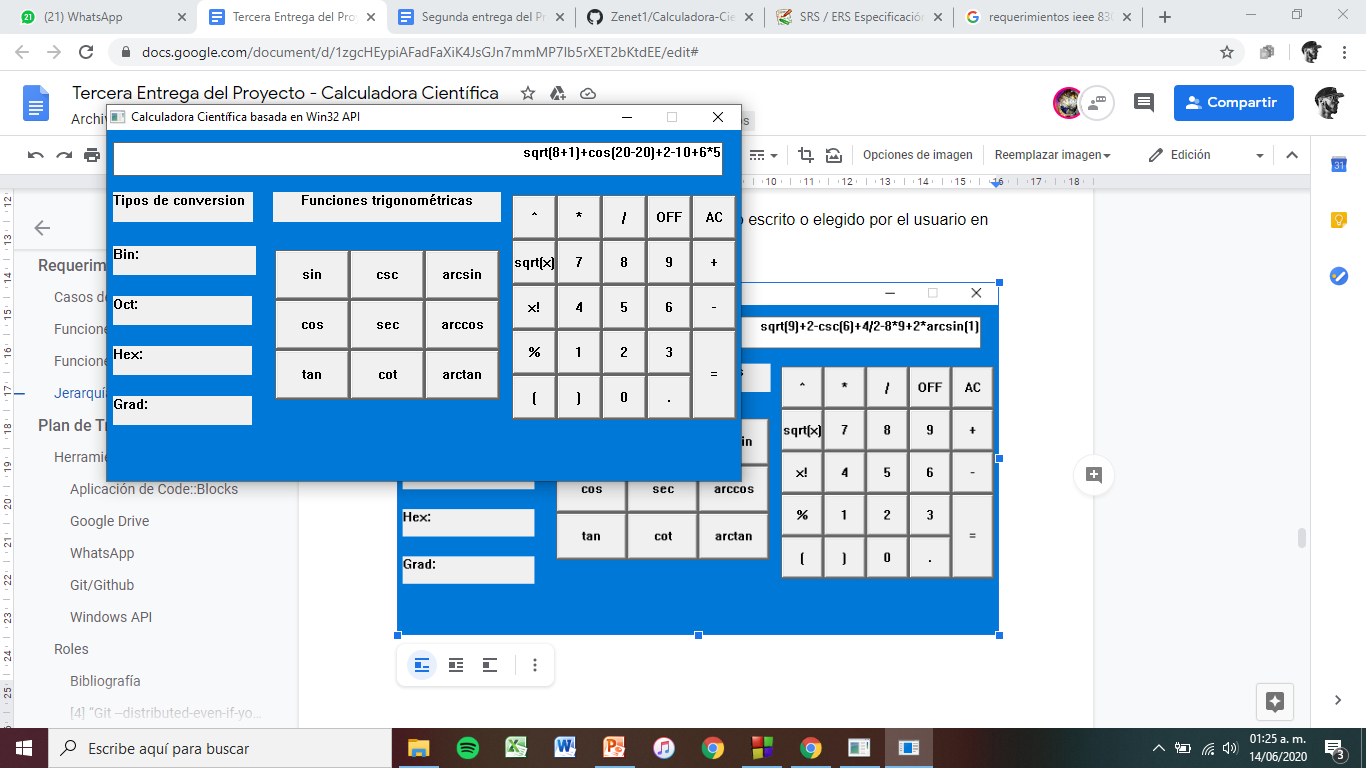
**Evento del sistema:** El sistema mostrará el resultado en la caja de texto, reemplazando la expresión anteriormente escrita, al mismo tiempo que mostrará las conversiones (binario, hexadecimal, octal y en grados) del mismo en el “Área de conversiones” de la calculadora.

## Jerarquía de operaciones

*Paso 1: Ingresar la entrada de operaciones.*

**Descripción:** El usuario deberá ingresar una o las operaciones matemáticas deseadas válida en la línea de texto (respetando el límite de 30 caracteres), utilizando ya sea el teclado físico o los botones implementados en la interfaz gráfica.

**Evento del sistema:** El sistema irá agregando lo escrito o elegido por el usuario en la caja de texto presentada.



Antes de presentar el resultado, el sistema calculará él mismo resolviendo la expresión basándose en la prioridad de cada uno de los operadores. A continuación se presentarán cada uno de ellos desde mayor hasta menor prioridad:

1. ***Paréntesis.***

Se tomará como primera línea de prioridad aquellas operaciones encerradas entre paréntesis, sin importar de qué operación se trate (sin embargo, dentro de los paréntesis, la prioridad de operadores será la misma).

En el ejemplo, se tomará como prioridad resolver todo lo que esté dentro del paréntesis de la función ‘sqrt’ y ‘cos’, que sería (8+1) y (20-20), respectivamente, quedando sqrt(9) y cos(0).

1. ***Funciones antes de un paréntesis.***

La segunda línea de la prioridad serán aquellas funciones que estén precedidas a un paréntesis, es decir, que después de la operación haya “(x)”; entiéndase todas las funciones trigonométricas así como su inversa, y raíz cuadrada. El usuario decidirá cuándo cerrar el paréntesis.

En el mismo ejemplo, coseno de (0) quedará como 1, lo cual se evaluará con el resto de la entrada, al igual que sqrt(9) quedando como 3.

1. ***Factorial, potencia y porcentaje.***

En tercer lugar, se le dará prioridad a las operaciones factorial de un número (x!), potencia (x^y) y porcentaje (x%=x/100)..

1. ***Multiplicación y división.***

La cuarta línea de prioridad consiste en los operadores aritméticos multiplicación (x\*y), así como división (x/y).

En el ejemplo, notamos que hay una multiplicación hasta el final, en este caso, 6\*5, por lo que se procederá a calcular la multiplicación tomando los números antes y después de la misma, en este caso, 6\*5 = 30.

1. ***Suma y resta.***

Por último lugar, se le dará prioridad a los operadores suma de dos números (x+y), y resta de dos números (x-y).

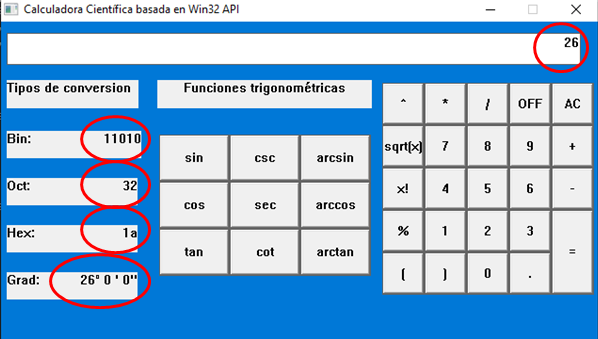
Como es de esperarse, en el ejemplo únicamente dichos operadores: ‘3+1+2-10+30’, por lo que se procederá a resolverlo de izquierda a derecha, quedando finalmente ‘26’.

*Paso 2: Obtener resultado.*

**Descripción:** El usuario deberá desplegar el resultado de la operación ingresada previamente, presionando el botón *“Igual a”* (=) en el botón implementado en la interfaz gráfica.

**Evento del sistema:** El sistema mostrará el resultado en la caja de texto, reemplazando la expresión anteriormente escrita, al mismo tiempo que mostrará las conversiones (binario, hexadecimal, octal y en grados) del mismo en el “Área de conversiones” de la calculadora.

En este caso, la última expresión del ejemplo es ‘sqrt(9)’ (*raíz cuadrada de 9*), la cual es igual a 3, con ‘coseno (0), aunado al resultado de suma y resta;siendo éste el resultado final que se mostrará “26”, así como su equivalente en binario, octal, hexadecimal y grados en sus respectivas etiquetas.



# 10. Plan de Trabajo

## Herramientas

A continuación se presentan las aplicaciones y herramientas que se usarán para llevar a cabo el proyecto, así como la descripción y propósito de cada uno.

### **Aplicación de Code::Blocks**

**[1]** Code::Blocks es un IDE (Entorno de Desarrollo Integrado) gratuito de C, C++ y Fortran, creado para satisfacer las necesidades más exigentes de sus usuarios. Está diseñado para ser muy extensible y totalmente configurable.

El propósito de su uso para este proyecto es para la escritura de código del programa, siendo que la aplicación proporciona herramientas para editar, compilar, depurar y ejecutar programas, en el cual se utilizará el lenguaje de programación **C.**

### **Google Drive**

**[2]** Google Drive es un servicio de almacenamiento externo brindado por la compañía Google; con el fin de respaldar y almacenar datos importantes, permitiendo guardar archivos de forma segura y abrirlos o editarlos desde cualquier dispositivo.

Su uso será para, tal y como se indica en su descripción anterior, una forma de almacenar los documentos y archivos creados, siendo una forma eficiente para trabajar de forma simultánea el equipo, a la vez que se guardará automáticamente la información agregada y cambios realizados.

### **WhatsApp**

**[3]** Es una aplicación que ofrece mensajería y llamadas, así como ser capaz de enviar y recibir variedad de archivos, entre ellos textos, fotos, videos, documentos y ubicación, entre otros.

Su propósito es simple, y es que servirá como medio de comunicación entre el equipo para una mejor organización sin importar la distancia o tiempo.

### **Git/Github**

**[4]** Git es un sistema de control de versiones gratuito y abierto diseñado para manejar desde pequeños hasta grandes y complejos proyectos con rapidez y eficiencia.

**[5]** Por otra parte, Github es una plataforma de alojamiento de código para el control de versiones y la colaboración entre miembros de trabajo. Permite no sólo a una, sino a varias personas trabajar juntos en proyectos desde cualquier lado.

Ambas herramientas las usaremos para recopilar todos los documentos, fuentes, y más importante, el proyecto principal para así, ser capaces de trabajar de forma separada y juntar todo cuando sea necesario, sin necesidad de realizar todo en una sola computadora.

De igual forma, sirve como respaldo y tener la ventaja de ver y comparar los cambios que se vayan realizando cada que se actualice o cambie el archivo principal.

Link del repositorio: <https://github.com/RicGrimaldo/Calculadora-Cient-fica>

### **Windows API**

**[6]** Win32 API (también llamada Windows API o simplemente Win API) es una plataforma nativa para las aplicaciones de Windows. Esta Interfaz de Programación de Aplicaciones se puede usar en todas las aplicaciones de escritorio, y las mismas funciones generalmente son compatibles con Windows de 32 bits y 64 bits.

Usaremos dicha plataforma para que, junto al lenguaje C escrito en la aplicación *Codeblocks,* seamos capaces de diseñar una GUI visualmente atractiva para el usuario para que el mismo sea capaz de interactuar con el programa.

## Roles

En esta sección se especificará el rol que tomará cada uno de nuestros integrantes a lo largo del proyecto:

* Ricardo Grimaldo: Líder, programador.
  + Realización de la función ‘Conversión a binario’.
  + Realización de la función ‘Conversión a hexadecimal’.
  + Diseño y programación de la GUI con Winapi.
  + Implementación de todas las funciones al programa principal, así como su anexo con respecto a la GUI.
  + Realización de validación y errores (límite, sintáctico y léxico).
* Edgar Bezares: Programador.
  + Realización de las funciones trigonométricas normales, recíprocas e inversas. (seno, coseno, tangente, cosecante, secante, cotangente, arcoseno, arcocoseno, arcotangente).
  + Realización de la función ‘Conversión a grados’.
  + Realización de la función ‘Conversión a octal’.
  + Elaboración de la evaluación infija a postfija.
* Eduardo López: Programador.
  + Implementación de la jerarquía de operaciones.
  + Realización todas las operaciones matemáticas.
  + Realización del cálculo de operaciones.
  + Realización de validación de errores matemáticos.
  + Evaluación de la postfija para obtener el resultado.

11. Bibliografía

**[1]** *“The open source, cross platform, free C, C++ and Fortran IDE.”* (2017). Recuperado de: <http://www.codeblocks.org/>

**[2]** *“Google Drive”*, (2012). Recuperado de: <https://www.google.es/intl/es_ALL/drive/>

**[3]** *“Acerca de Whatsapp”,* (2009). Recuperado de: <https://www.whatsapp.com/about/>

**[4]** *“Git* *--distributed-even-if-your-workflow-isnt”,* (s.f). Recuperado de: <https://git-scm.com/>

**[5]** *“Github Guides - Hello World”,* (2016). Recuperado de: <https://guides.github.com/activities/hello-world/>

**[6]** *“API Index for desktop Windows applications”,* (2019). Recuperado de: <https://docs.microsoft.com/en-us/windows/win32/apiindex/api-index-portal>

[7] “*SRS / ERS Especificación de requerimientos de software”* , (2010). Recuperado de <https://www.icesi.edu.co/departamentos/tecnologias_informacion_comunicaciones/proyectos/lisa/home/analisis/srs/srs#:~:text=El%20est%C3%A1ndar%20IEEE%20830%2D1998,el%20grupo%20de%20desarrollo%20para>