

Indice

Datos del estudiante.....	2
Proyecto inicial.....	2
Sincronizar el proyecto local con un repositorio remoto.....	3
Realizando traducciones.....	4
Realizando funcionalidades extra.....	5
Conversor Celsius ↔ Fahrenheit.....	6
Conversor Pascal ↔ Bar.....	6
Conversor Gramos ↔ Kilogramos.....	7
Conversor Centímetros ↔ Metros.....	7
Resultado final en producción.....	8

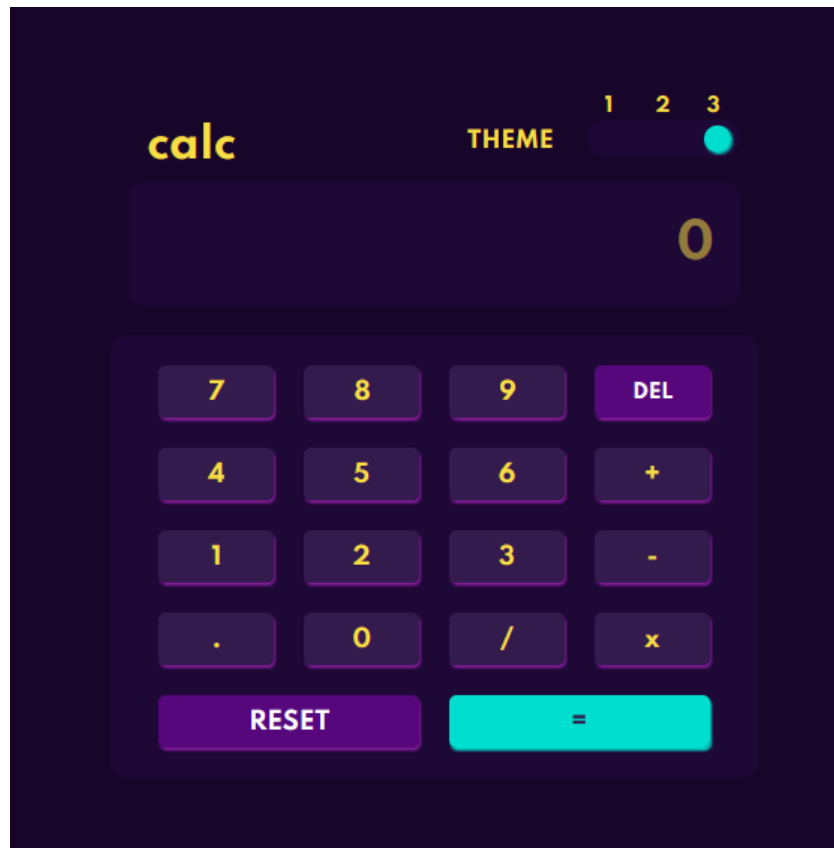
Datos del estudiante

El taller fue realizado por **Luis Danilo Juajinoy Gálvez** con código estudiantil **2130341182**

Proyecto inicial

Como base para este taller se toma la calculadora del repositorio

<https://github.com/Igorcbraz/Calculadora.git>, en el cual se tiene una calculadora inicial como en la siguiente imagen

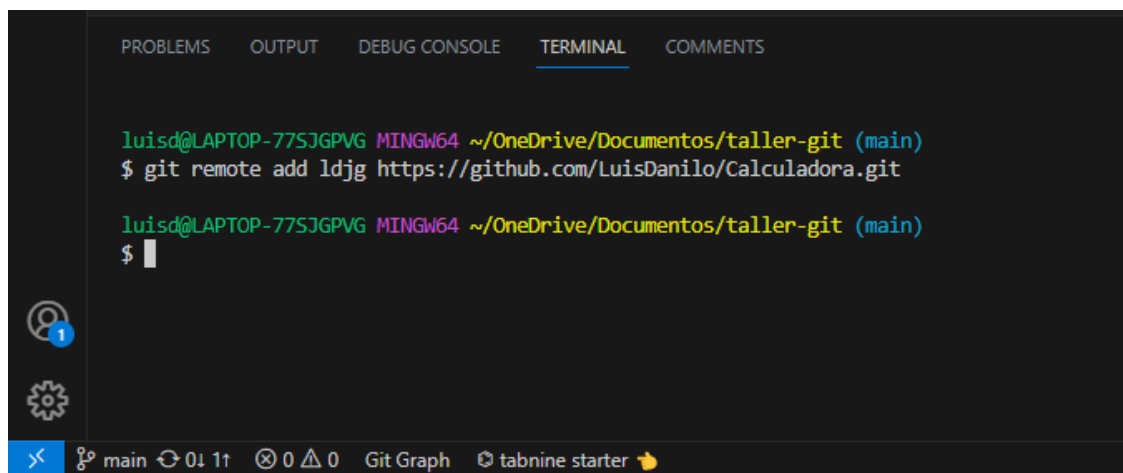


De este modo se logra tener una copia del mismo de manera local, lista para el desarrollo.

Sincronizar el proyecto local con un repositorio remoto

Para comenzar a subir los cambios requeridos en un repositorio remoto aparte del repositorio original se creó un repositorio en la cuenta personal del estudiante, en este caso se crea el repositorio <https://github.com/LuisDanilo/Calculadora.git>.

Una vez creado se procede a sincronizar el proyecto local con el repositorio previamente creado, para no tener confusiones entre el repositorio remoto original y el repositorio remoto del estudiante se nombra al último como **ldjg** de modo que se realiza de la siguiente manera:



```
luisd@LAPTOP-77SJGPVG MINGW64 ~/OneDrive/Documentos/taller-git (main)
$ git remote add ldjg https://github.com/LuisDanilo/Calculadora.git

luisd@LAPTOP-77SJGPVG MINGW64 ~/OneDrive/Documentos/taller-git (main)
$
```

The screenshot shows a terminal window within a code editor. The terminal title bar includes tabs for PROBLEMS, OUTPUT, DEBUG CONSOLE, TERMINAL (which is active), and COMMENTS. The terminal output shows the user 'luisd' at 'LAPTOP-77SJGPVG' in a 'MINGW64' environment, located at '~/OneDrive/Documentos/taller-git' on the 'main' branch. The command executed is 'git remote add ldjg https://github.com/LuisDanilo/Calculadora.git'. Below this, the prompt '\$' is shown again, indicating the command has completed. The bottom status bar of the editor shows 'main', a refresh icon, '0 11', a close icon, '0 0', 'Git Graph', and 'tabnine starter'.

Realizando traducciones

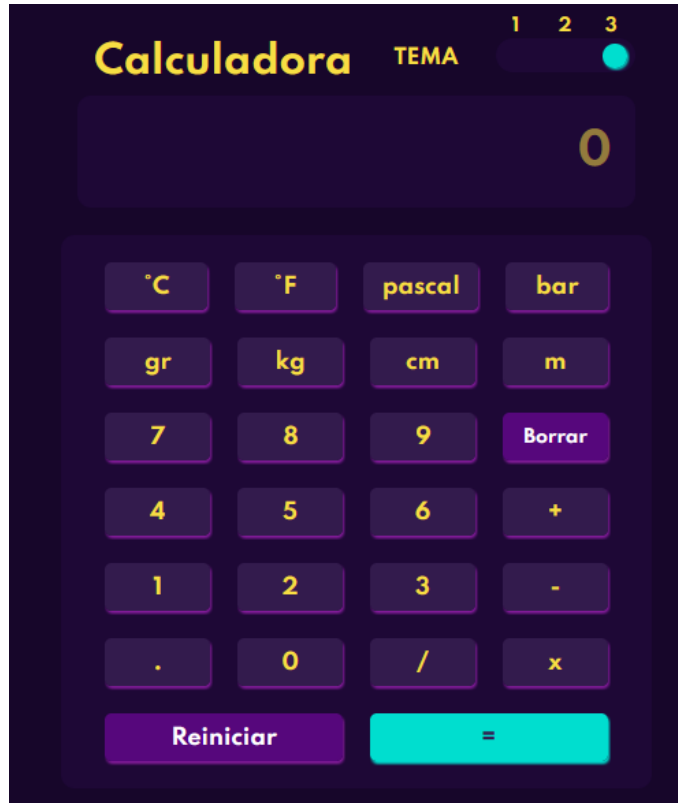
Una de las tareas consiste en escribir las traducciones necesarias en la calculadora y realizar dichos cambios en la rama *main*, de modo que la aplicación ahora luce como se muestra a continuación



Realizando funcionalidades extra



Con el objetivo de llevar registro de las nuevas funcionalidades se crea una nueva rama basada en la combinación de la primera letra del nombre y el primer apellido del estudiante. En este caso *ljuajinoy*

Para la construcción de las nuevas funcionalidades fue necesario agregar algunos elementos extras a la interfaz gráfica así como la logica en Javascript que soportara las mismas.





Al tratarse de una calculadora sencilla se reutilizó y modificó el código implementado por el autor original en el cálculo de las operaciones básicas para las diferentes conversiones, dado a que las conversiones solicitadas consisten en operaciones aritméticas sencillas



Conversor Celsius ↔ Fahrenheit

	<p>Converte la cifra digitada (se asume que en grados Fahrenheit) a grados celsius</p>	<pre> 26 // To celsius logic 27 function toCelsius() { 28 if (result.value != "") { 29 let result2 = result.value; 30 result.value = eval(`\${result2}-32)*(5/9)`); 31 } else { 32 alert("Error, ingrese valores válidos") 33 } 34 } </pre>
	<p>Converte la cifra digitada (se asume que en grados Celsius) a grados Fahrenheit</p>	<pre> 36 // To fahrenheit logic 37 function toFahrenheit() { 38 if (result.value != "") { 39 let result2 = result.value; 40 result.value = eval(`\${result2}*(9/5)+32`); 41 } else { 42 alert("Error, ingrese valores válidos") 43 } 44 } </pre>



Conversor Pascal ↔ Bar

	<p>Converte la cifra digitada (se asume que en bares) a pascales</p>	<pre> 46 // To pascal logic 47 function toPascal() { 48 if (result.value != "") { 49 let result2 = result.value; 50 result.value = eval(`\${result2}*100000`); 51 } else { 52 alert("Error, ingrese valores válidos") 53 } 54 } </pre>
	<p>Converte la cifra digitada (se asume que en pascales) a bares</p>	<pre> 56 // To bar logic 57 function toBar() { 58 if (result.value != "") { 59 let result2 = result.value; 60 result.value = eval(`\${result2}/100000`); 61 } else { 62 alert("Error, ingrese valores válidos") 63 } 64 } </pre>

Conversor Gramos ↔ Kilogramos

	Converte la cifra digitada (se asume que en kilogramos) a gramos	<pre>66 // To gr logic 67 ~function toGr() { 68 ~ if (result.value != "") { 69 ~ let result2 = result.value; 70 ~ result.value = eval(`\${result2}*1000`) 71 ~ } else { 72 ~ alert("Error, ingrese valores válidos") 73 ~ } 74 ~ }</pre>
	Converte la cifra digitada (se asume que en gramos) a kilogramos	<pre>76 // To kg logic 77 ~function toKg() { 78 ~ if (result.value != "") { 79 ~ let result2 = result.value; 80 ~ result.value = eval(`\${result2}/1000`) 81 ~ } else { 82 ~ alert("Error, ingrese valores válidos") 83 ~ } 84 ~ }</pre>

Conversor Centímetros ↔ Metros

	Converte la cifra digitada (se asume que en metros) a centímetros	<pre>86 // To cm logic 87 ~function toCm() { 88 ~ if (result.value != "") { 89 ~ let result2 = result.value; 90 ~ result.value = eval(`\${result2}*100`) 91 ~ } else { 92 ~ alert("Error, ingrese valores válidos") 93 ~ } 94 ~ }</pre>
	Converte la cifra digitada (se asume que en centímetros) a metros	<pre>96 // To m logic 97 ~function toM() { 98 ~ if (result.value != "") { 99 ~ let result2 = result.value; 100 ~ result.value = eval(`\${result2}/100`) 101 ~ } else { 102 ~ alert("Error, ingrese valores válidos") 103 ~ } 104 ~ }</pre>

Resultado final en producción

Como etapa final se llevan todos los cambios realizados en la rama ***ljuajinoy*** a la rama ***main***, asumida como la rama de producción.