# Exercici 1

## Introducció

El primer exercici serà la instal·lació del nostre entorn de treball per a poder crear aplicacions amb el framework Ionic. Aquest framework permet la creació d’aplicacions per a dispositius Android, iOS, Windows i navegadors al tractar-se d’uns framework multiplataforma.

Tot i que es tracta d’un framework multiplataforma pot incorporar codi nadiu dels tres sistemes operatius amb relativa facilitat per a garantir que l’aplicació es podrà executar a cada tipus de terminal.

## NodeJS i NPM

El primer que haurem de fer serà instal·lar les aplicacions NodeJS i NPM. Per fer-ho anirem a la [pàgina web de NodeJS](https://nodejs.org/es/) i descarreguem la opció recomendada. Un cop descarregada l’aplicació executarem l’instal·lador.

Quan tinguem l’aplicació instal·lada obrirem un terminal i escriurem el següent:

$node-v

Al fer-ho es retornarà la versió de NodeJS i ens assegurarem de que la instal·lació s’ha realitzat de forma correcta i sense cap tipus de problema.

## Ionic

Per a instal·lar Ionic obrirem el terminal i introduirem la següent línia de comandament:

$npm install –g @ionic/cli

La instal·lació d’Ionic durpa uns quants minuts Durant els quals s’ens anirà mostrant tots els paquets que s’estan instal·lant al nostre equip. Quan la instal·lació finalitzi escriurem el següent comandament per a veure si tot s’ha instal·lat de forma correcta al nostre equip.

$ionic –v

S0ens retornarà la versió d’Ionic per línia de comandament. En cas contrari voldrà dir que la instal·lació no s’ha realitat de forma correcta.

## Visual Code

Per a facilitar la creració del codi és recomanable instal·lar un editor de text. EN aquest cas s’ha triat el IDE Visual Code al tenir inserit un terminal que permet executar ordres de comandament per a anar escalant i desenvolupant les aplicacions.

Aquest IDE es gratuït i es pot descarregar desde la seva [pàgina web](https://code.visualstudio.com/download). El IDE és gratuït i no cal que ens registrem per a poder utilitzar-ho. En cas de que es volgui, podem instal·lar qualsevol altre IDE com a podria ser Atom o un editor de text. En aquest últim cas es recomana l’editor [Notepad++](https://notepad-plus-plus.org/downloads/) que també és gratuït.

## Creant una carpeta

Un cop instal·lat tot l’entorn crearem una carpeta a la nostre unitat que contendrà totes les aplicacions que anem desenvolupant. Es recomana la creació d’aquesta carpeta per a tenir tots els nostres projectes localitzats. La creació de la carpeta la podem fer desde l’explorador de Windows o desde un terminal amb el comandament

$mkdri nom-carpeta

Veurem con ens apareix una nova carpeta.

## Creació d’una aplicació

Per a crear la nostre primera aplicació obrirem un terminal i escriurem:

$ionic start firstApp blank -–type=angular

Aquest comandament ens crearà un esquelet buit d’una aplicació que hem anomenatn firstApp. Un cop creada l’estructura de l’aplicació escriure,

$cd firstApp

Per accedir a la carpeta de l’aplicació i un cop a dins escriurem

$code .

Aquest útim comandament ens obrirà el nostre projecta a Visual Code. Aquesta primera aplicació es pot descarregar de forma gratuïta des del [repositori de github de bcnitb](https://github.com/BCNITB/Ionic-excercises/tree/main/exercici-01/firstApp).

# Exercici 2

## Introducció

En aquest segon exercici farem una petita aplicació on el jugador haurà de calcular el resultat d’una operació matemàtica amb dos nombres que es generaran de forma aleatòria per la aplicació.

Així doncs crearem una aplicació anomenada **guessOperation** per a començar a fer l’aplicació. L’esquelet de l’aplicació es pot descarregar des del [repositori de bcnitb](https://github.com/BCNITB/Ionic-excercises/tree/main/exercici-02/guesOperation).

## Descripció de l’aplicació

L’aplicació que acabem de descarregar és l’esuqelet de la futura aplicació. Per a veure l’aspecte actual de l’aplicació obrirem un terminal i escriurem el següent comandament:

$ionic serve

Al cap d’uns minuts s’ens obrirà el navegador d’internet que tenim per defecte i ens mostrarà l’aplicació com si fos una aplicació web.

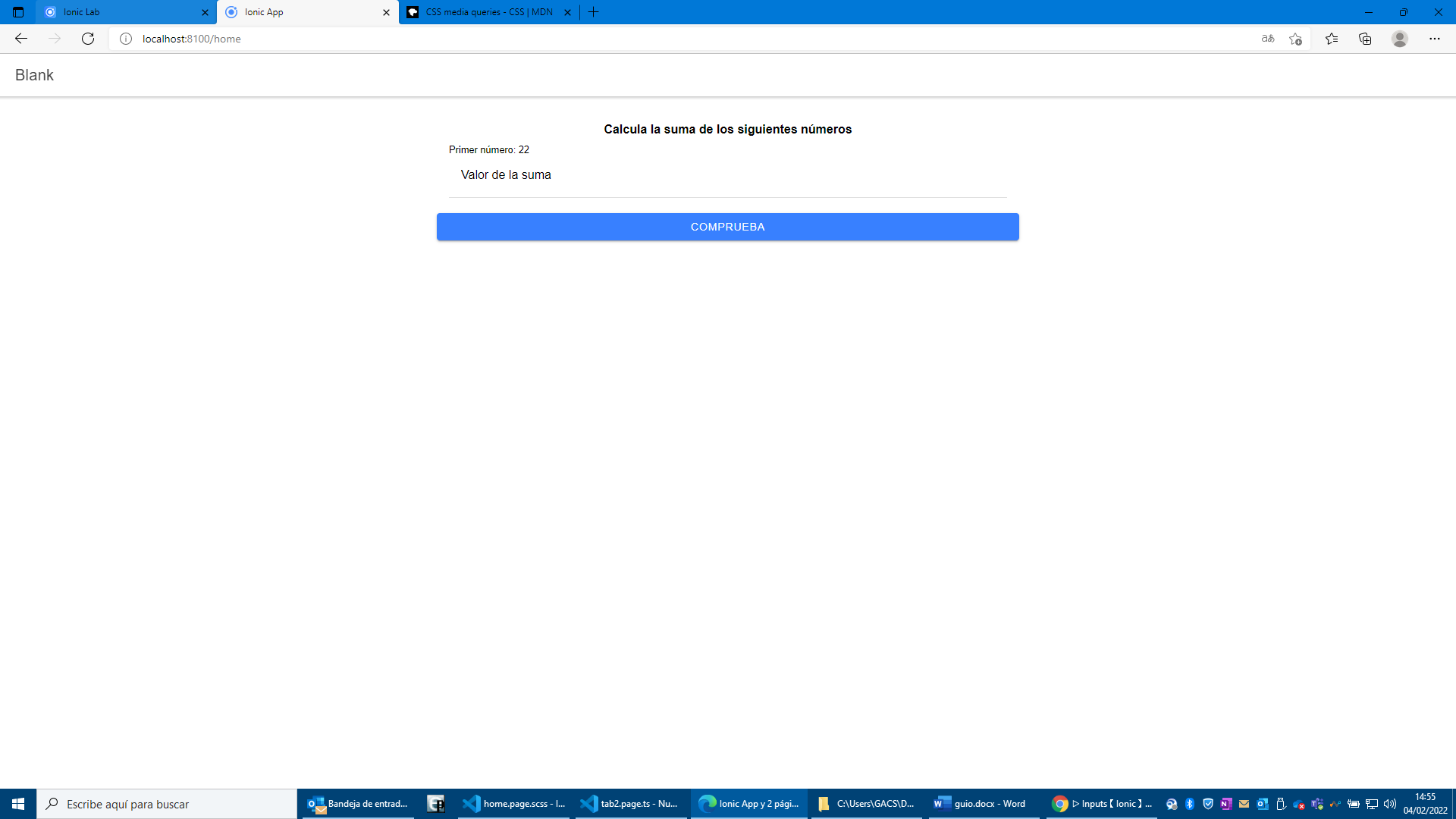


Figura 1. Aplicació web.

En cas de que volguem veure la seva apariència a un dispositiu mòbil pitjarem les tecles **Control + C** i escriurem el següent comandament escriurem:

$ionic serve –l

Es tornarà a obrir un navegador d’internet però veurem l’aplicació dins d’un terminal iOS i un Android.

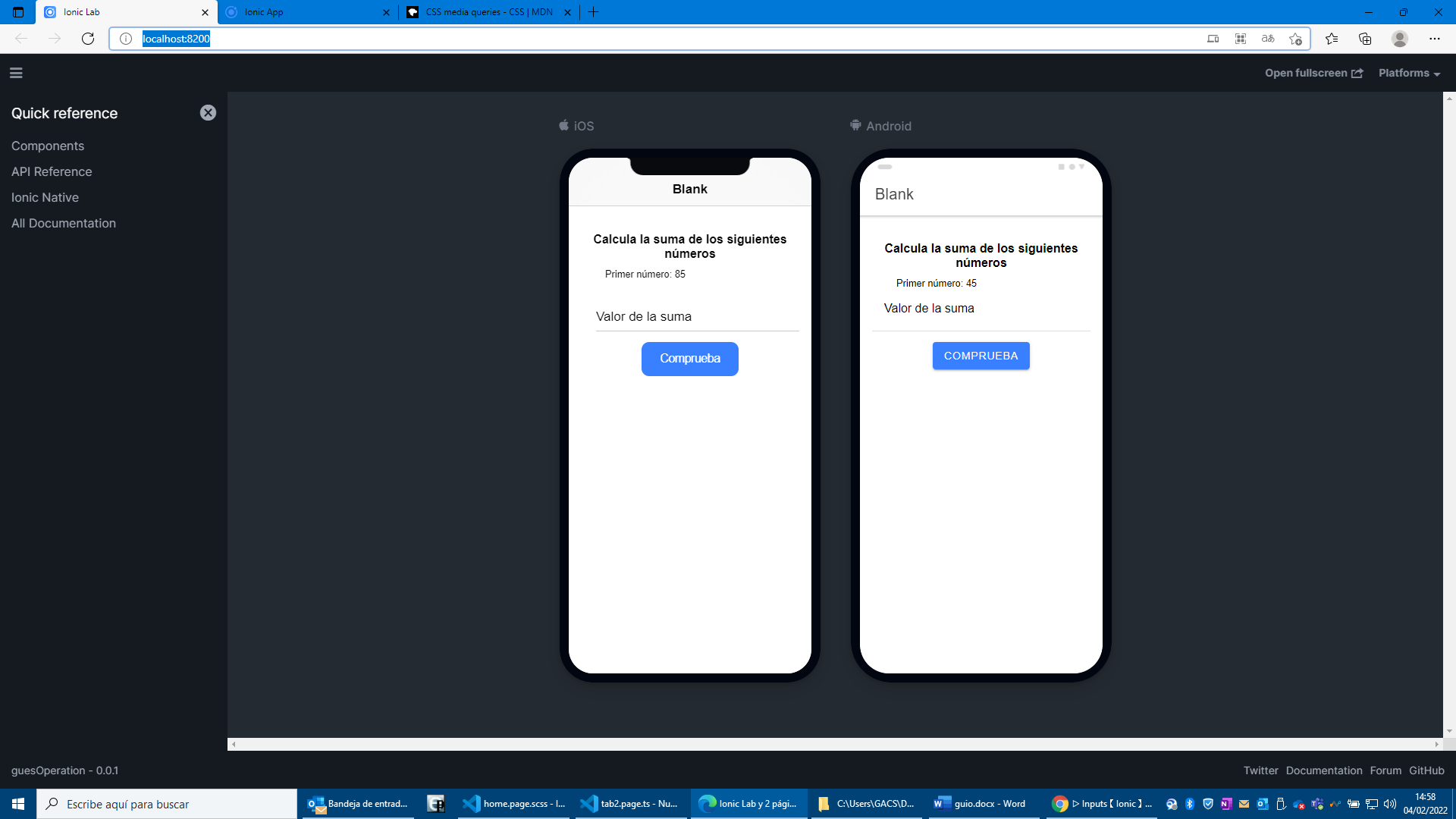


Figura 2. Aplicació en dispositius mòbils.

## Conceptes

Si s’obre l’arxiu [**home.page.html**](http://home.page.html), dins la carpeta **src/app/home** del projecte, es pot observar el codi en **HTML** de la vista. En aquest codi es pot veure que hi ha una etiqueta d’encapçalament **<h2>**, una de paràgraf **<p>** i un altre de formulari **<form>**, bàsicament. I dins la etiqueta de formulari hi ha una **<label>** i una d’un camp d’edició **<ion-input>**.

## Exercici

Sota de la etiqueta de paràgraf hi ha un comentari amb el text TODO. El que es demana és que es substitueixi aquesta etiqueta per a que el mostri el següent text:

Segon número:

I un número aletatòri que ha de ser diferent del número anterior. El resultat es mostra a la Figura 3.

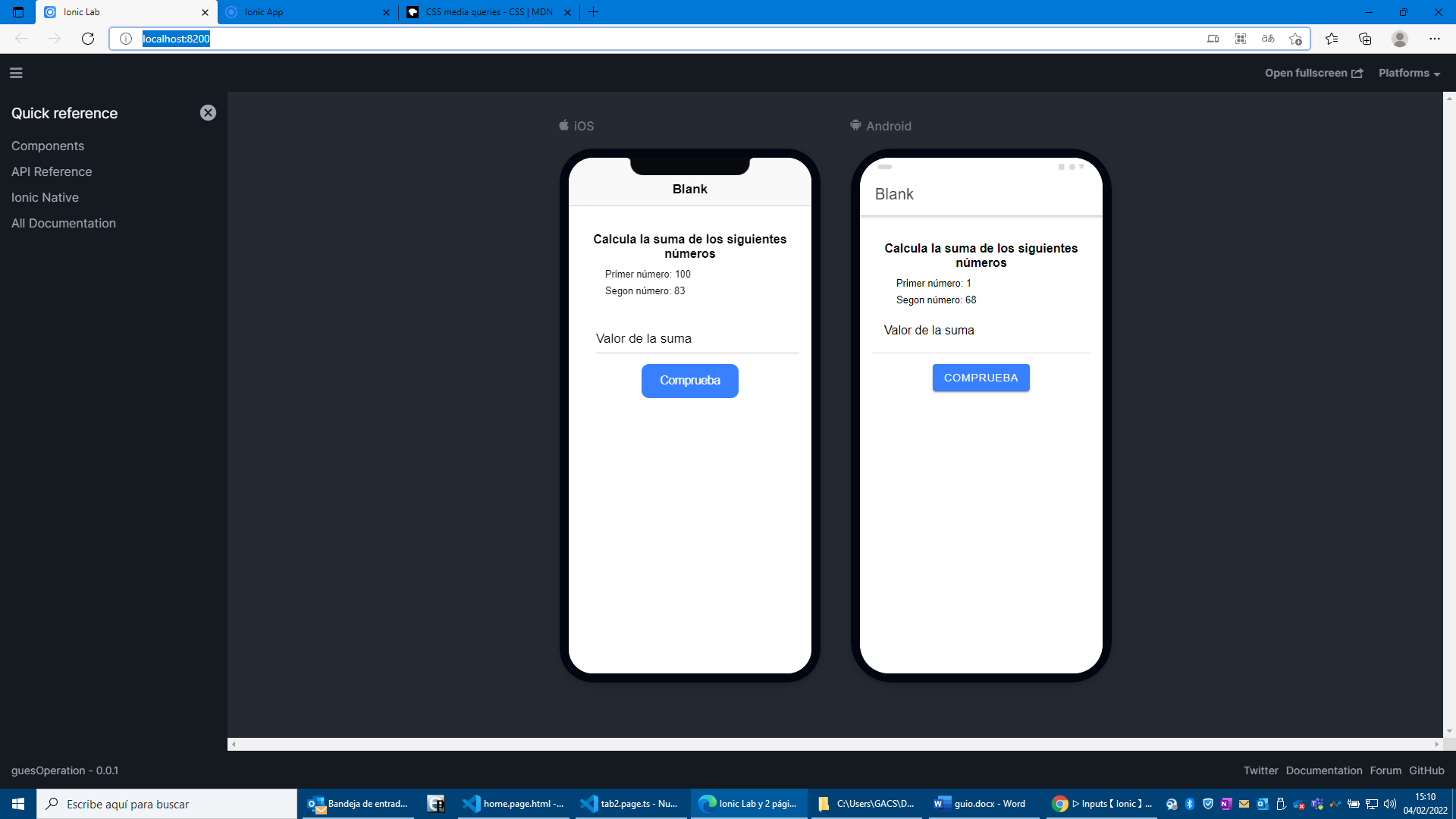


Figura 3. Solució de l'exercici.

# Exercici 3

## Introducció

En aquest tercer exercic traballarem amb codi de TypeScript per a retocar la lògica de l’aplicació per a obtenir una resposta o una altre depenent de la resposta de l’usuari. La lògica de l’aplicació està al fitxer [**home.page.ts**](http://home.page.ts)que es pot trobar a la mateixa direcció que el fitxer que hem vist a l’exercici 2, **src/app/home**.

Per a poder fer l’exercic haurem de descarregar la carpeta execici 3 que trobarem al [repositor de bcnitb](https://github.com/BCNITB/Ionic-excercises/tree/main/exercici-03/guesOperation). Si executem l’aplicació veurem que inicialment no hi ha cap modificació, però si introduïm un valor i premem el botó **Comproba** obtenim una o una altre resposta depenent de si el nostre resultat es correcte o no. Si introduïm el resultat correcte obtenim unua confrimació a la pantalla tal i com es mostra aFigura 4. Mentres que si el resultat que introduïm es inferior al correcte obtindrem el missatge de la figura Figura 5.

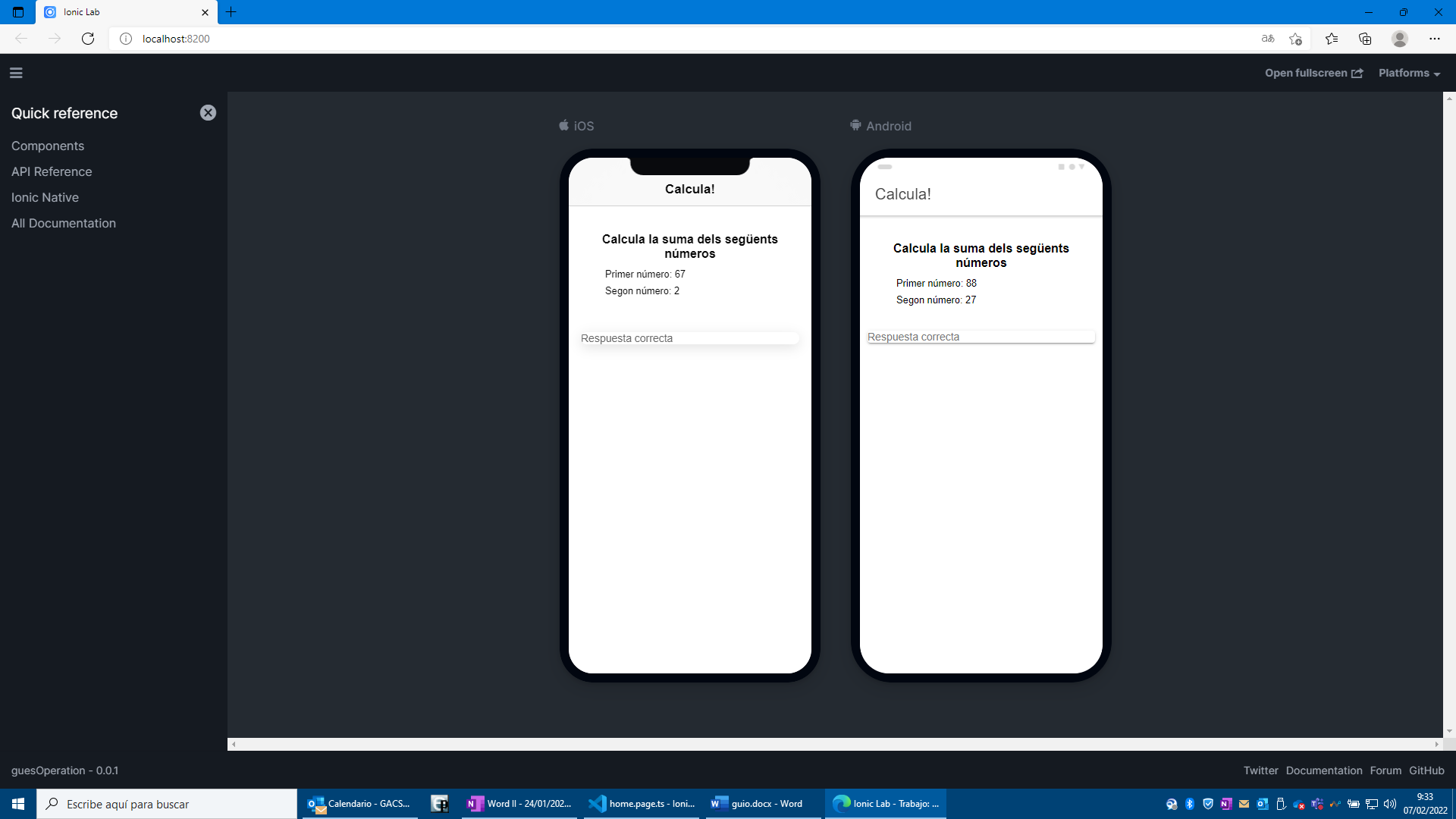


Figura . Resultat correcte.

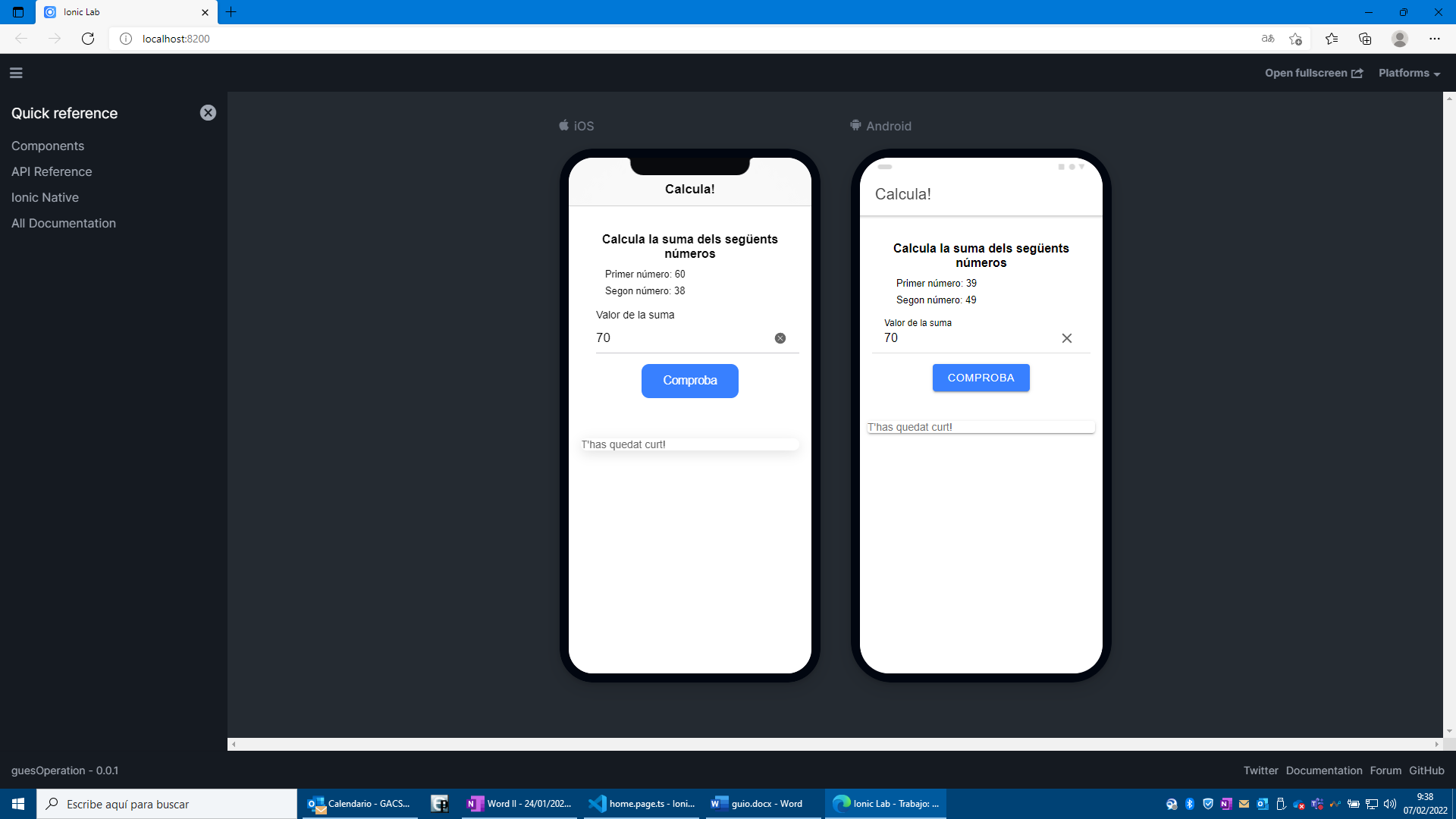


Figura . Missatge si la suma és superior al valor introduït.