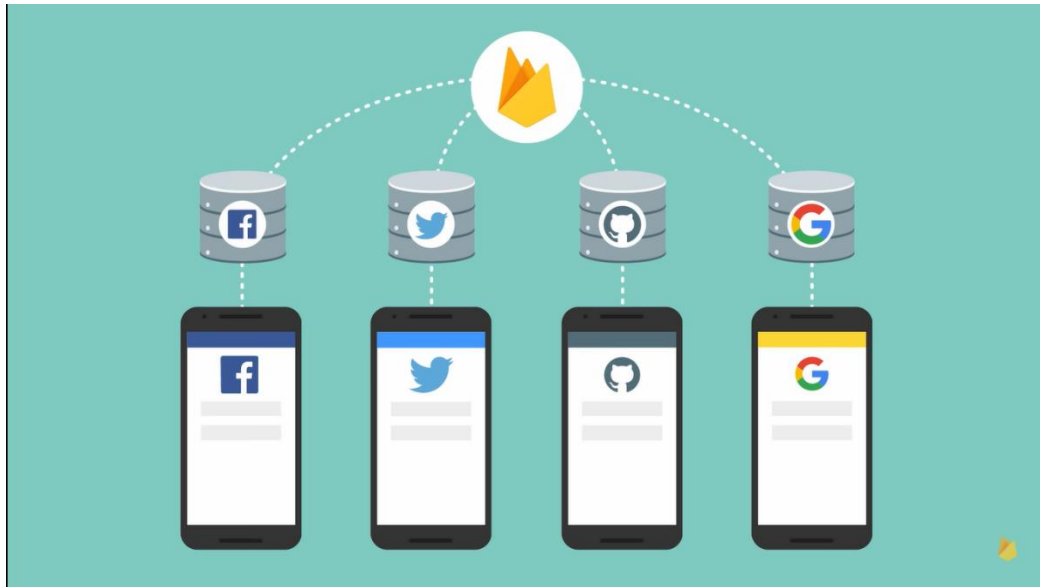
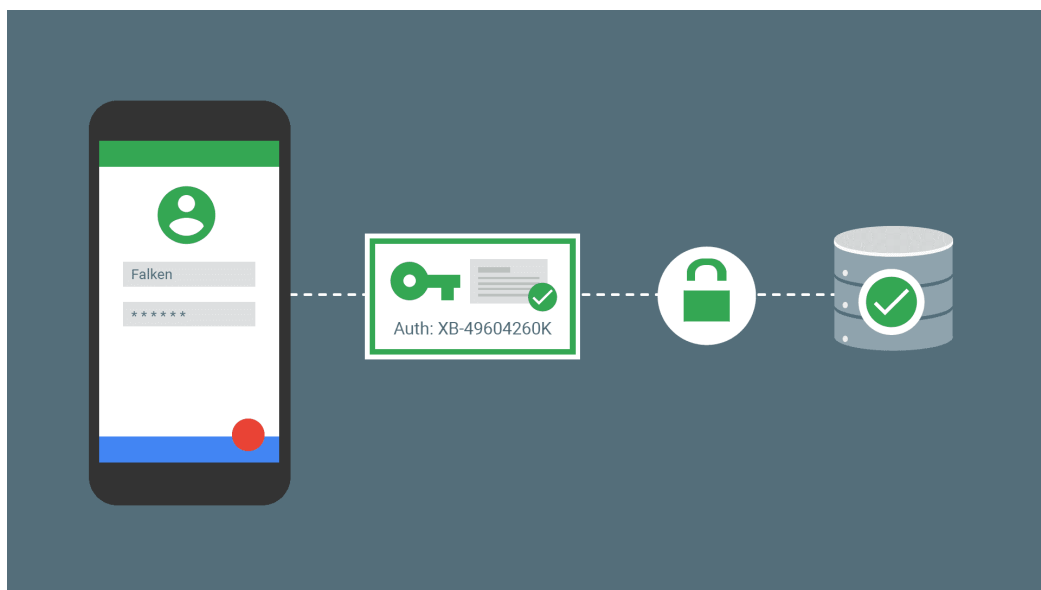


## Use of a single sign-on service with an authorization framework

Provides an authentication solution, through flows managed by a user access UI. It has an Auth component implemented which is used for websites and mobile devices that can maximize access conversion. Through this authentication you have a better security since you can get relevant data that are accessible with verified emails.



It has a comprehensive security through Google, Smart lock and by the password manager in Chrome.



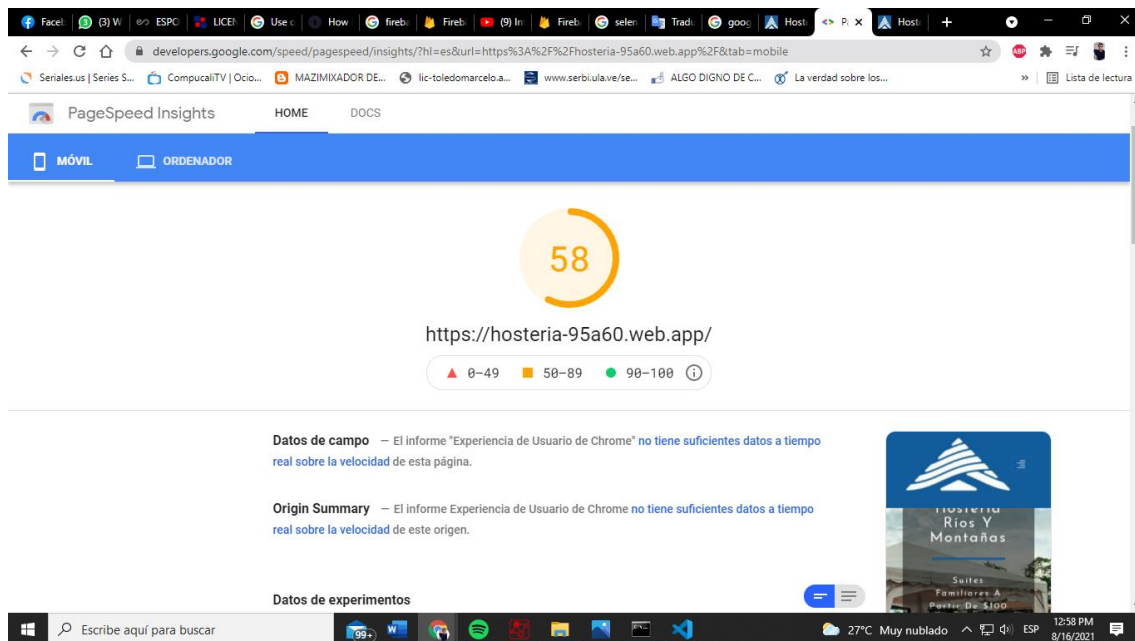
Our code validates the emails, if they exist it returns a token with that information.

```
const login = React.useCallback(async () => {
  try {
    const res = await auth.signInWithEmailAndPassword(email, pass)
    console.log(res.user)
    setEmail('')
    setPass('')
    setError(null)
    props.history.push('/')
  } catch (error) {
    console.log(error)
    if(error.code === 'auth/invalid-email'){
      setError('Email no válido')
      setEmail('')
      setPass('')
    }
    if(error.code === 'auth/user-not-found'){
      setError('Email no registrado')
      setEmail('')
      setPass('')
    }
    if(error.code === 'auth/wrong-password'){
      setError('Contraseña incorrecta')
      setEmail('')
      setPass('')
    }
  }
}, [email, pass, props.history])
```

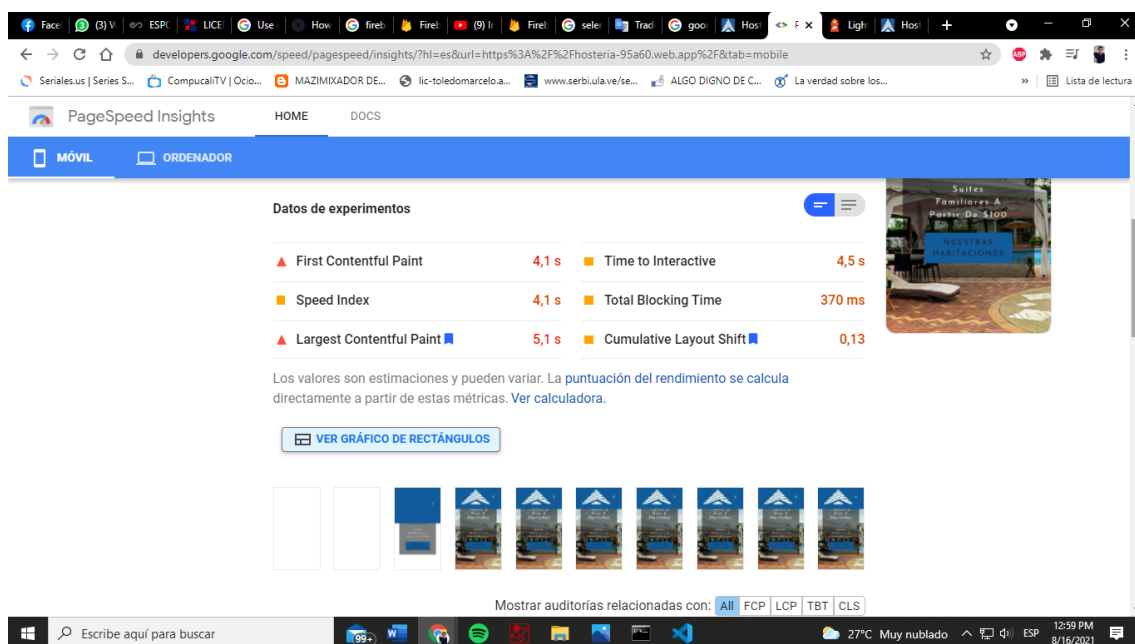
Through this function we can validate registered users in our application and thanks to the comprehensive security you can reset the password by means of "firebase" where a link is sent to the end user's email and if you wish you can proceed to make the change.

## Extra-Use of a GUI test automation tool

It has been used "google developers", is a tool created for developers with many functions, with which we have tested some windows of our page.



For the mobile and web part the load test has medium speed in load time from a mobile application, in the web part a certain similarity was shown in the load test.



The app gives us a lot of data, such as data from experiments and some recommendations for diagnosis by web and mobile.

PageSpeed Insights

HOME DOCS

MÓVIL ORDENADOR

Oportunidad Ahorro estimado

- ▲ Reduce el contenido JavaScript que no se use 1,17 s
- Elimina los recursos que bloqueen el renderizado 0,59 s

Diagnósticos — Consulta más información sobre el rendimiento de tu aplicación. Estos datos no **afectan directamente** a la puntuación del rendimiento.

- ▲ Asegúrate de que el texto permanece visible mientras se carga la fuente web
- ▲ Publica recursos estáticos con una política de caché eficaz — Se han encontrado 7 recursos
- Evita encadenar solicitudes críticas — Se han encontrado 6 cadenas
- Medidas y marcas de User Timing — 1 tiempo de usuario
- Reduce el número de solicitudes y el tamaño de las transferencias — 11 solicitudes · 591 KIB
- Renderizado del mayor elemento con contenido — 1 elemento encontrado
- Evitar cambios de diseño importantes — 2 elementos encontrados
- Evita tareas largas del hilo principal — 3 tareas largas encontradas

After applying the appropriate modifications to the code, the app improves its loading performance.

PageSpeed Insights

HOME DOCS

MÓVIL ORDENADOR

97

https://hosteria-95a60.web.app/

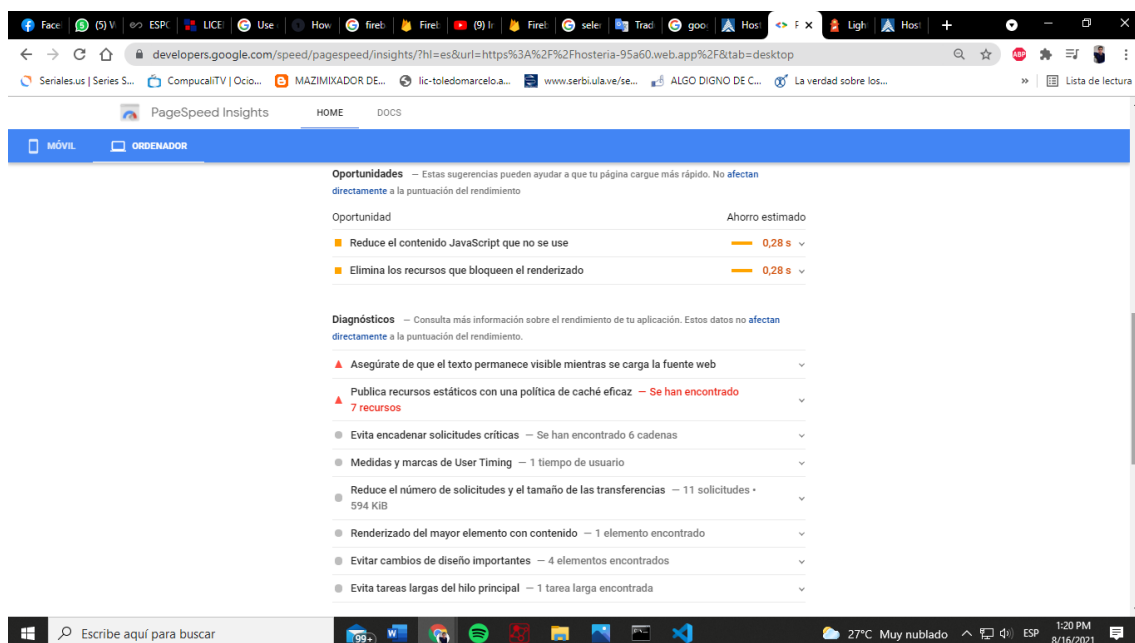
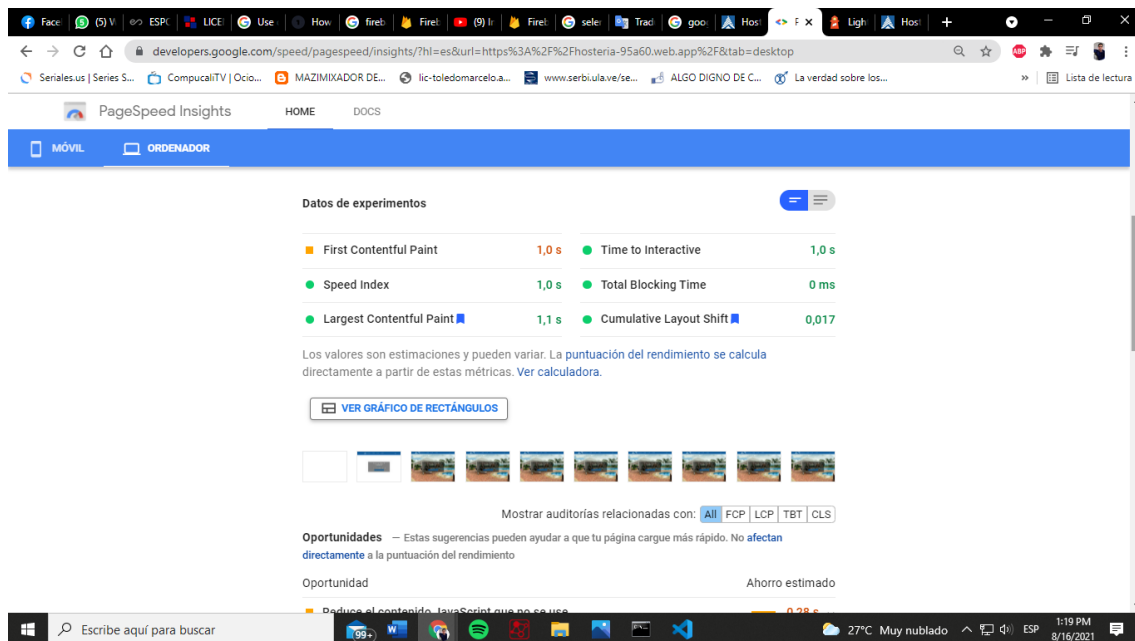
▲ 0-49 ■ 50-89 ● 90-100 ⓘ

Datos de campo — El informe "Experiencia de Usuario de Chrome" no tiene suficientes datos a tiempo real sobre la velocidad de esta página.

Origin Summary — El informe Experiencia de Usuario de Chrome no tiene suficientes datos a tiempo real sobre la velocidad de este origen.

Datos de experimentos

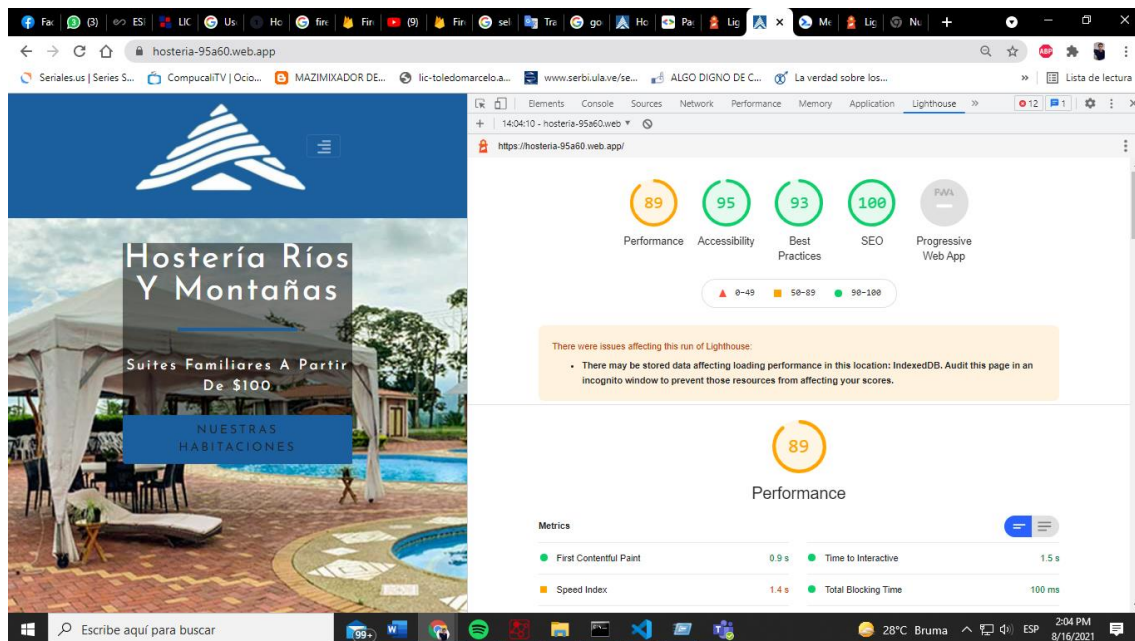
- First Contentful Paint 1,0 s
- Time to Interactive 1,0 s



Source : <https://developers.google.com/speed/pagespeed/insights/?hl=es>

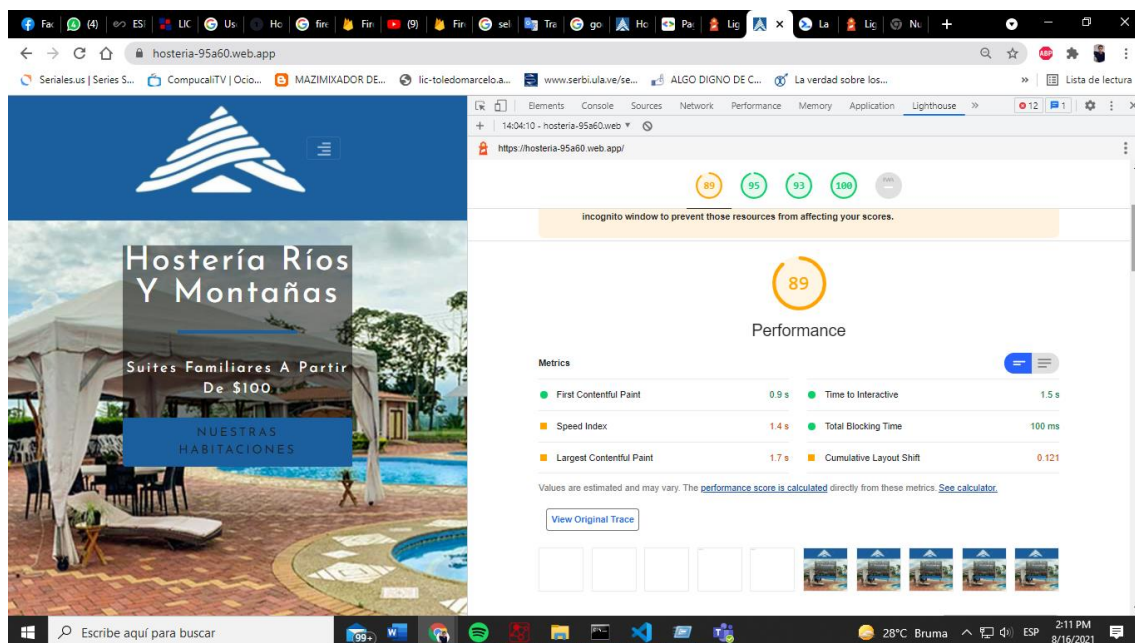
## Extra-Load Testing

With the same "Google developers" but now we use "Lighthouse" with all its features. To access this Google feature or tool we must place ourselves on the website you want to analyze, then press inspect and in the bar on the right side where it says: elements, console, source and others. Search lighthouse, select the categories to analyze and the type of device. To finish just click on generate report and wait.



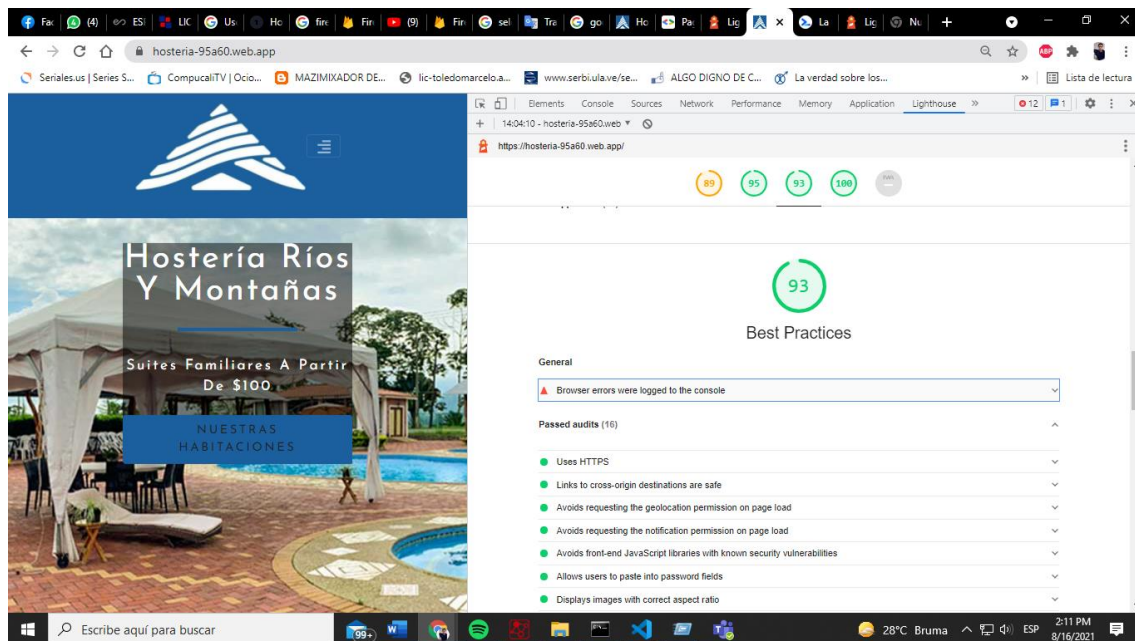
With improved code, we can see how performance is in range. To be extremely optimal, you should reach more than 90 in each section.

Performance is the interaction time to determine the load delay, in other words, the load time.

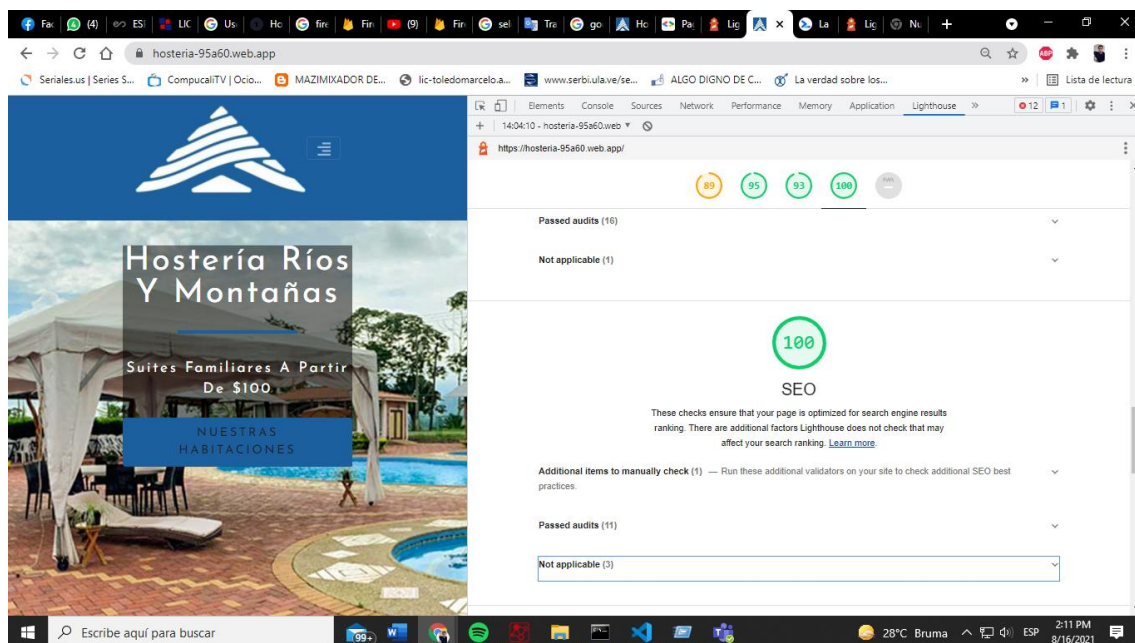


In **best practices**, rate the use of HTTPS within the entire render of the page.





In **SEO** verifies the quality of the code or implementation for a better positioning on the web.



**Accessibility** checks for common issues that might prevent multiple users from accessing content.

