

## Prueba - Quién ganará

- Para realizar esta prueba debes haber estudiado previamente todo el material disponibilizado correspondiente al módulo.
- Desarrollo prueba:
  - La prueba se debe desarrollar de manera Individual.
  - Para la realización de la prueba necesitarás apoyarte del archivo *Apoyo Prueba - Quién ganará*

### Descripción

¿Quién ganará? es una lotería que transmiten por televisión nacional tradicionalmente todos los martes por la noche, y a pesar de ser muy conocido en el país, las ventas han bajado considerablemente debido a que la única forma de registrarse en una partida es dirigiéndose presencialmente a una de las sucursales de la empresa, lo cual provocó molestia por parte de los usuarios, ya que ahora exigen que se disponibilice un sistema web para poder jugar por internet.

En esta prueba, pondrás en práctica tus conocimientos de Node, al tener que construir un servidor que atienda las peticiones de una aplicación cliente disponible en el Apoyo Prueba - ¿Quién ganará?, que incluye la aplicación de cliente y los archivos JSON iniciales.

El servidor debe:

- Almacenar usuarios nuevos
- Devolver todos los usuarios almacenados
- Devolver la información correspondiente al premio
- Modificar la información correspondiente al premio
- Generar y devolver al ganador entre los participantes almacenados
- Enviar un correo electrónico masivo a todos los participantes anunciando quién fue el ganador (Se recomienda incluir tu correo personal para recibir el anuncio del ganador)

La siguiente imagen es una referencia de la prueba terminada.

### Bienvenido a ¿Quién ganará?

Participantes [Agregar Participante](#)

id	Correo	Nombre	País	Foto
e05b67	haroen.stoops@example.com	Mr Haroen Stoops	Netherlands	
e15db5	molly.white@example.com	Miss Molly White	United Kingdom	
a7ac95	antoine.williams@example.com	Mr Antoine Williams	Canada	
f22297	felix.ennis@example.com	Mr Felix Ennis	Canada	
5877eb	emma.chow@example.com	Ms Emma Chow	Canada	
01d7f9	fred.foster@example.com	Mr Fred Foster	United States	
217872	levi.jackson@example.com	Mr Levi Jackson	New Zealand	
604e4e	jeremy.chan@example.com	Mr Jeremy Chan	Canada	
a17c05	kristiane.myrland@example.com	Ms Kristiane Myrland	Norway	
a7a3b0	peniel.teixeira@example.com	Mr Peniel Teixeira	Brazil	

#### Un viaje todo pagado a la Playa

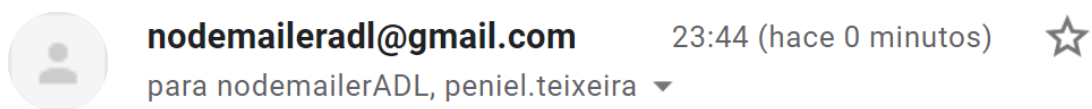
[Editar Premio](#)

[¡Generar ganador!](#)

¡Felicidades!  
**Mr Felix Ennis**

¡¡¡Ganaste!!!

Una vez definido el ganador, debes mandar un correo electrónico a todos los participantes anunciando quién fué el ganador. Como se muestra en la siguiente imagen:



**Anuncio: El ganador de ¿Quién ganará? fue Mr Felix Ennis, gracias a todos por participar**

Rutas que debes crear en tu servidor:

- **/ GET:** Debe devolver el documento HTML disponibilizado en el apoyo
- **/usuario POST:** Almacena un usuario
- **/usuarios GET:** Devuelve todos los usuarios almacenados
- **/premio GET:** Devuelve los datos del premio
- **/premio PUT:** Edita los datos del premio
- **/ganador GET:** Devuelve al ganador y envía los correos electrónicos

## Requerimientos

1. Devolver el documento HTML disponibilizado en el Apoyo Prueba - ¿Quién ganará? a través de la ruta raíz **GET /** del servidor.
2. El botón "Agregar participante" de la aplicación cliente genera una petición POST (sin payload) esperando que el servidor registre un nuevo usuario random con la API [randomuser](#), por lo que debes preparar una ruta **POST /usuario** en el servidor, que ejecute una función asíncrona importada de un archivo externo al del servidor (la función debe ser un módulo), para obtener la data de un nuevo usuario y la acumule en un JSON (usuarios.json). El objeto correspondiente al usuario que se almacenará debe tener un id generado con el paquete UUID.
3. El cliente luego de agregar un participante y al momento de cargar el sitio web, le solicita al servidor todos los usuarios almacenados, por lo que deberás disponibilizar una ruta **GET /usuarios** que devuelva todos los usuarios almacenados en el servidor (usuarios.json).
4. La aplicación cliente al ser cargada, ejecuta una consulta al servidor pidiendo la información referente al premio de la partida, por lo que deberás disponibilizar una ruta **GET /premio** que devuelva los datos del premio almacenados en el servidor (premio.json).
5. La aplicación cliente tiene un botón para modificar la información correspondiente al premio, por lo que deberás disponibilizar una ruta **PUT /premio** que reciba el payload de la consulta y modifique los datos almacenados en el servidor (premio.json).
6. La aplicación cliente ejecuta a través de un botón una función que genera una consulta al servidor esperando obtener el nombre y la foto del ganador.

En este último requisito deberás crear una ruta **GET /ganador** que procese una lógica que seleccione de forma aleatoria un participante para definir al ganador, además de utilizar el paquete nodemailer para generar un envío de correo electrónico masivo a todos los participantes anunciando el nombre del participante ganador.

- Incluye tu correo electrónico en este envío para que recibas el mail correspondiente.
  - La lógica correspondiente al envío de correos debe estar en un archivo diferente e importado en el servidor como un módulo
7. Crear una ruta **GET /{id}** que devuelva según el parámetro id de un usuario la data del participante almacenado en el archivo usuarios.json.