



## Proyecto 2

---

---

Pedro Ricardo Gordillo González  
Carlos Alexis Ovalle de León

202031683  
202031064



## Índice

<b>Índice.....</b>	<b>2</b>
<b>Plantilla historias de usuario.....</b>	<b>3</b>
<b>Prioridades.....</b>	<b>4</b>
<b>Historias de usuario.....</b>	<b>5</b>
Administrador/Ayudantes.....	5
1. Creación de la empresa -> 2.....	5
2. Edición de la empresa -> 2.....	5
3. Creación de roles (nombre) -> 2.....	6
4. Creación de permisos específicos y asignación con un rol -> 8.....	6
5. Actualización de roles (nombre y/o permisos asociados) -> 5.....	7
6. Creación de personal (posibilidad de agregar imágenes) -> 3.....	8
7. Manejo de personal -> 5.....	9
8. Actualización de información y rol del personal -> 5.....	10
9. Actualización de perfil personal (Contraseña e imagen si aplica) -> 3.....	11
10. Elegir si las reservas de citas se realizan de forma manual (sin balanceador) -> 8	12
11. Elegir si las reservas de citas son aleatorias (con balanceador) -> 21.....	12
12. Administración de reportes -> 13.....	14
13. Creación de elementos dentro de la plataforma -> 21.....	15
14. Administración de elementos dentro de la plataforma -> 21.....	17
15. Generación de facturas -> 8.....	18
16. Generación de comprobantes -> 8.....	19
17. Creación de jornada laboral.....	20
18. Edición de jornada laboral.....	21
19. Reprogramación de citas como cliente.....	21
20. Cancelaciones de citas -> 21.....	22
21. Login (jwt y a2f) y mostrar las opciones según el rol del cliente o administrador ->	24
55.....	24
22. Reprogramación de citas como administrador-> 55.....	24
Clientes.....	26
1. Registrarse en la plataforma -> 3.....	26
2. Recuperar contraseña -> 5.....	27
3. Visualización de calendario personal -> 21.....	28
4. Agendar citas -> 21.....	29
5. Cancelar cita -> 13.....	30
6. Visualización del historial de citas.....	31
7. Mensajes de notificaciones.....	32



8. Ver comprobantes y facturas.....	33
-------------------------------------	----



## Plantilla historias de usuario

ID: 1	Titulo	
Descripción		
Estimación:	Dependencias:	
Prioridad:		

ID:	Pruebas de aceptación	
Descripción		



## Prioridades



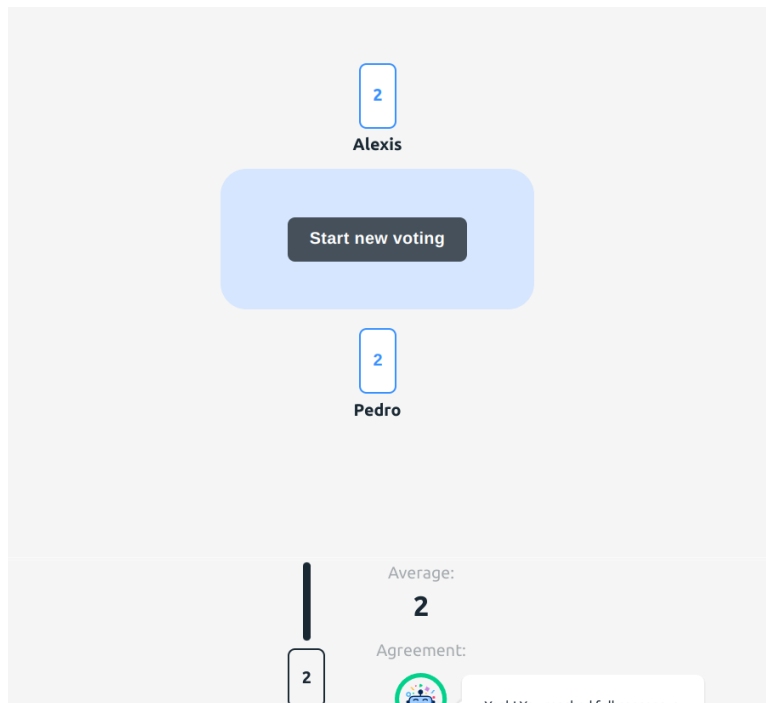


## Historias de usuario

[documentacion historias](#)

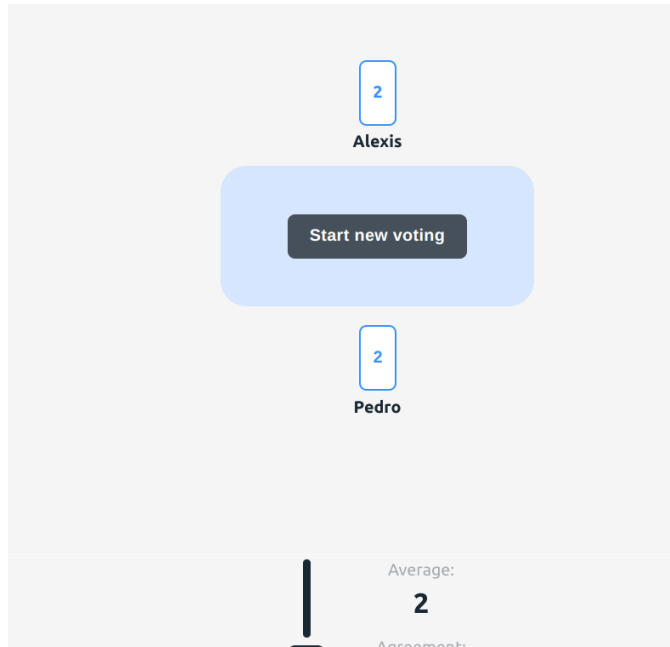
### Administrador/Ayudantes

#### 1. Creación de la empresa -> 2





## 2. Edición de la empresa -> 2



## 3. Creación de roles (nombre) -> 2

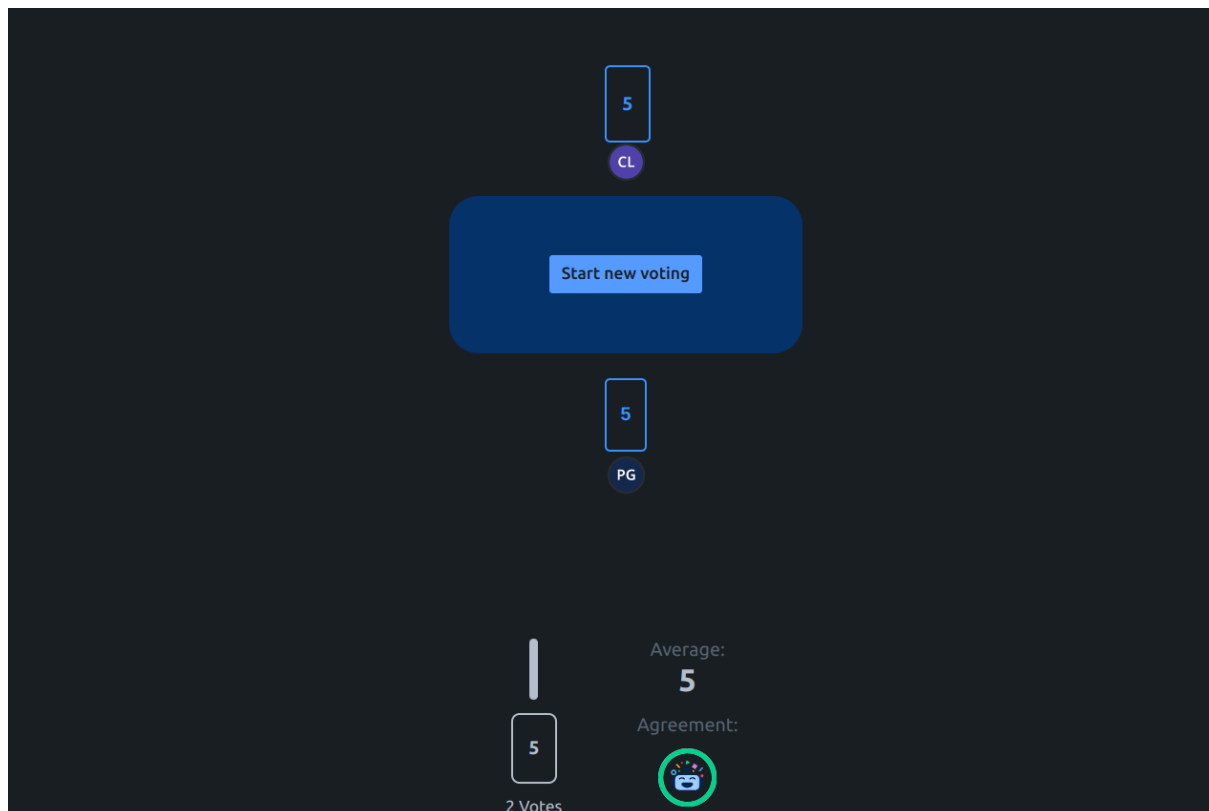




#### 4. Creación de permisos específicos y asignación con un rol -> 8



#### 5. Actualización de roles (nombre y/o permisos asociados) -> 5







## 6. Creación de personal (posibilidad de agregar imágenes) -> 3





## 7. Manejo de personal -> 5





## 8. Actualización de información y rol del personal -> 5

5

CL

Start new voting

5

PG

5

2 Votes

Average: 5

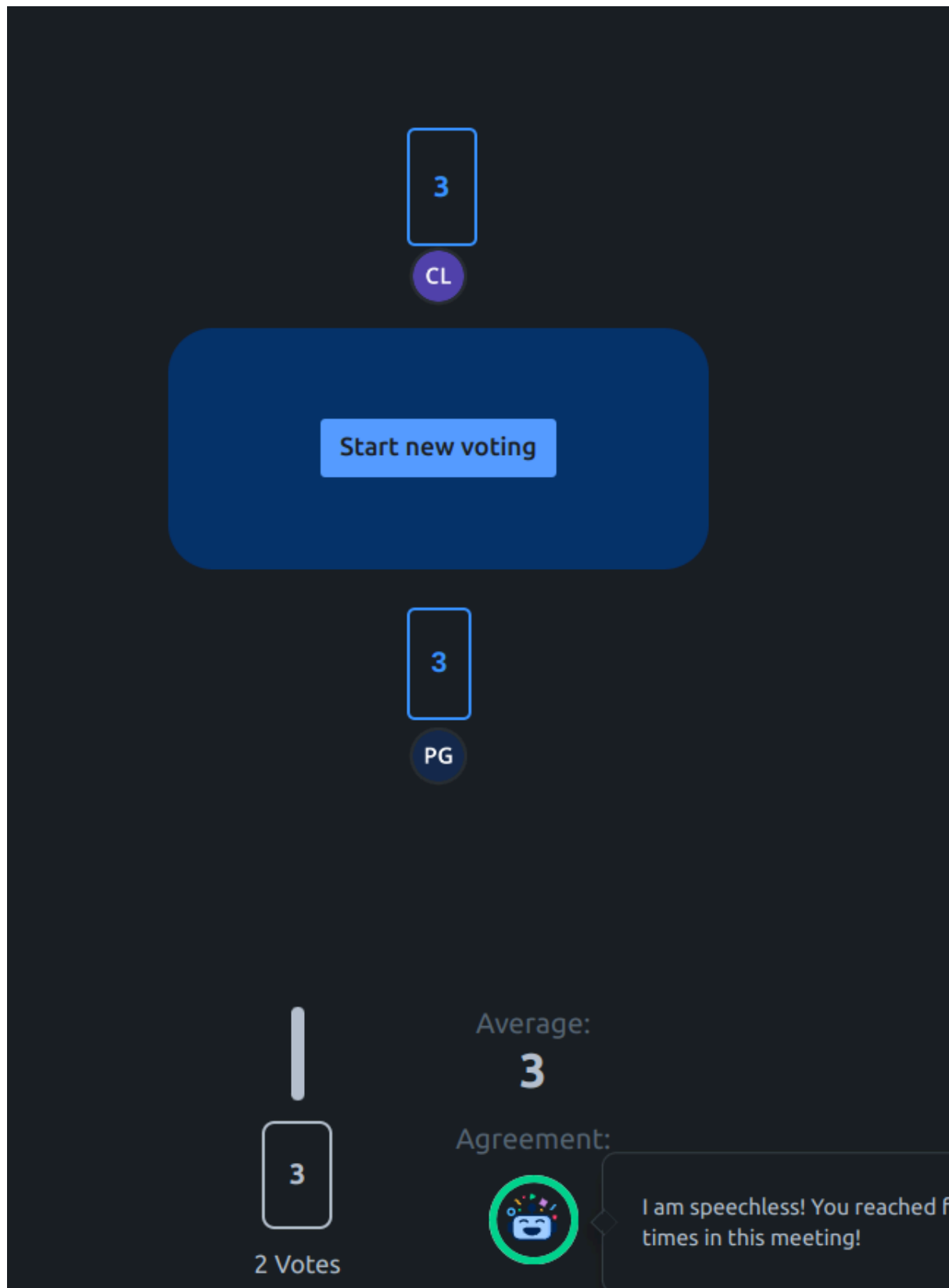
Agreement:

I am speechless! You reached full o times in this meeting!



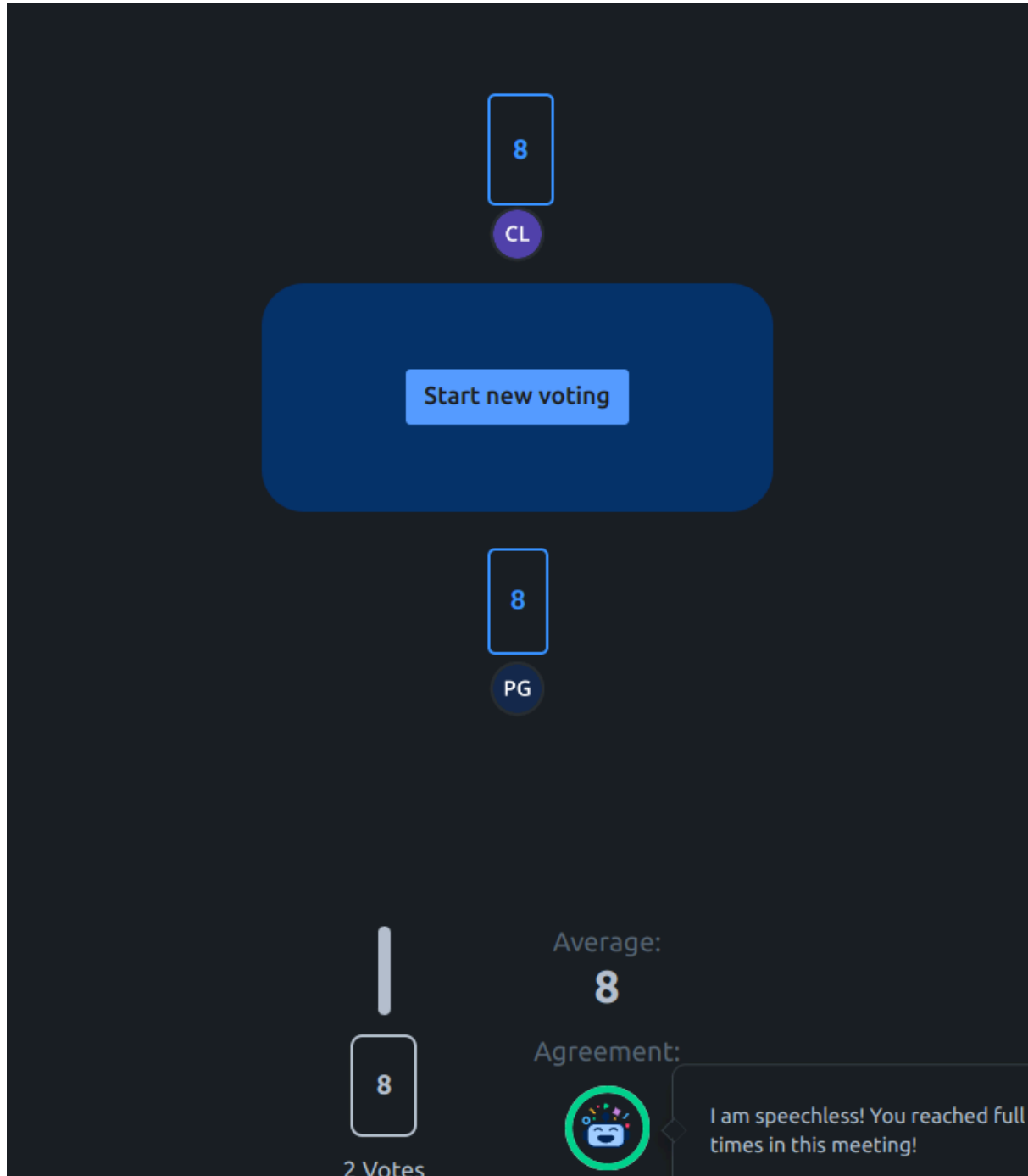
9. Actualización de perfil personal (Contraseña e imagen si aplica) ->

3





10. Elegir si las reservas de citas se realizan de forma manual (sin balanceador) -> 8

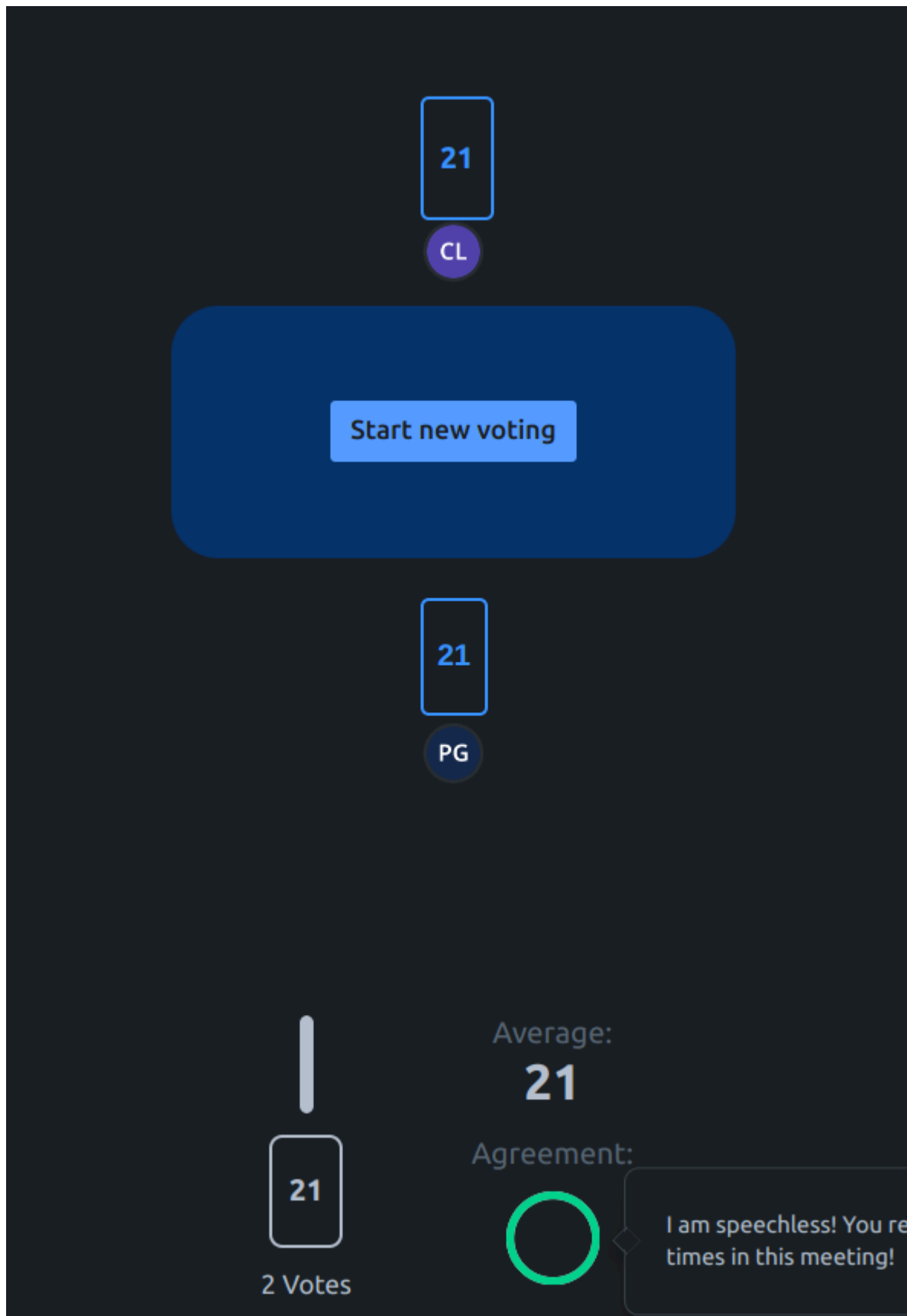


11. Elegir si las reservas de citas son aleatorias (con balanceador)  
-> 21

Universidad de San Carlos de Guatemala  
Centro Universitario de Occidente  
División de Ingeniería  
Ingeniería en Ciencias y Sistemas  
Análisis y Diseño de Sistemas 1



**USAC**  
TRICENTENARIA  
Universidad de San Carlos de Guatemala







## 12. Administración de reportes -> 13



13

CL

Start new voting

13

PG

13

2 Votes

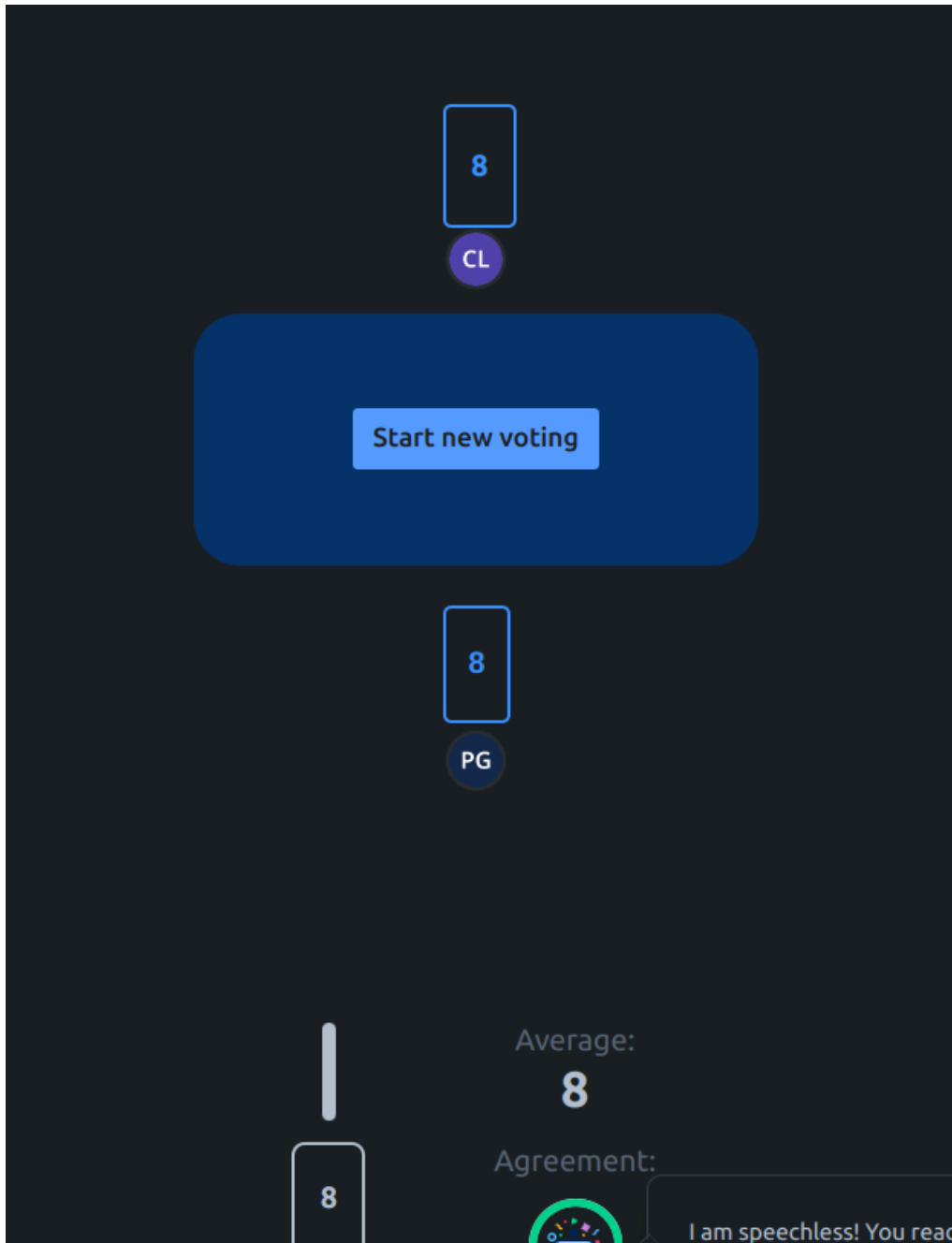
Average:  
13

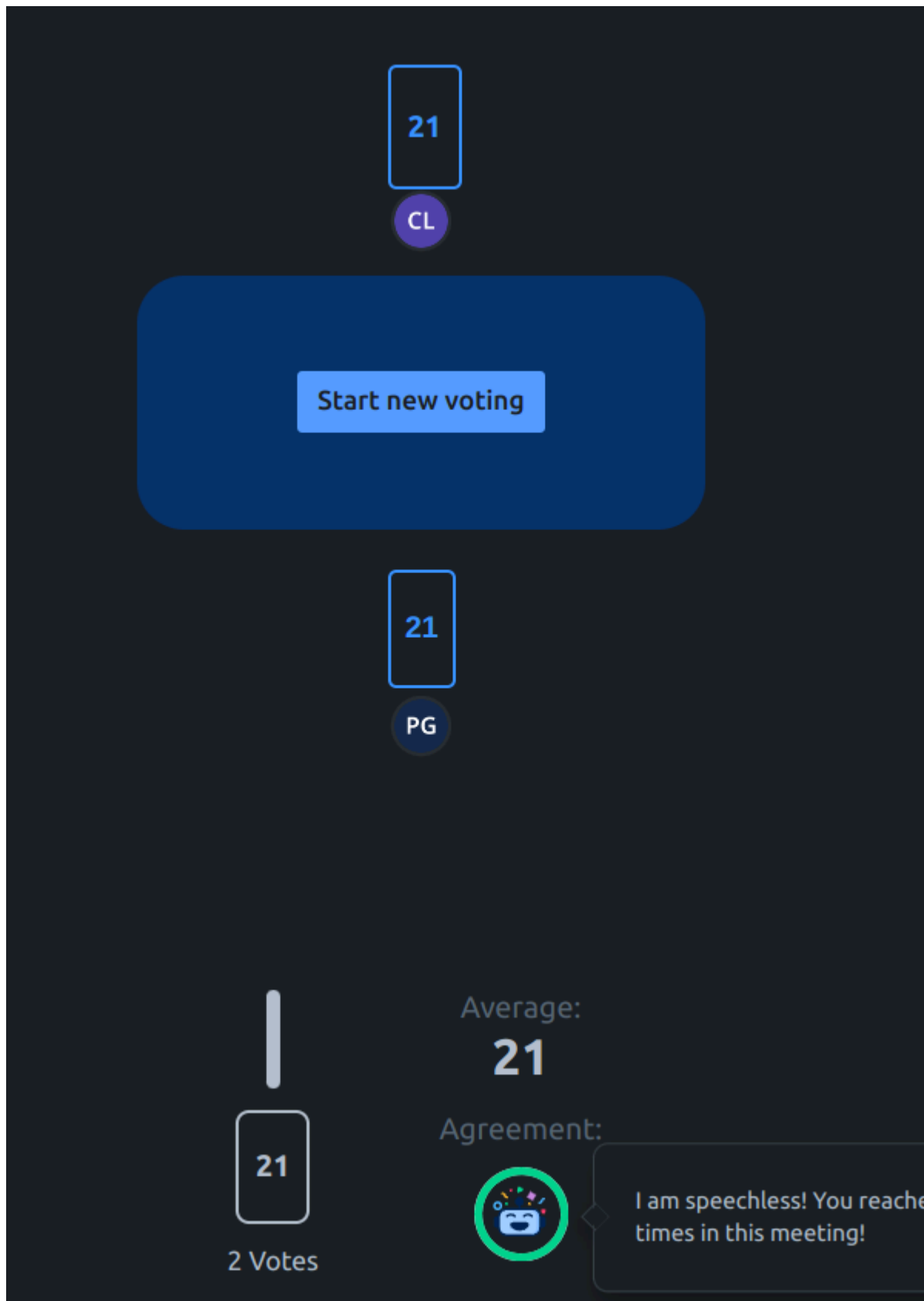
Agreement:

I am speechless! You r  
times in this meeting!



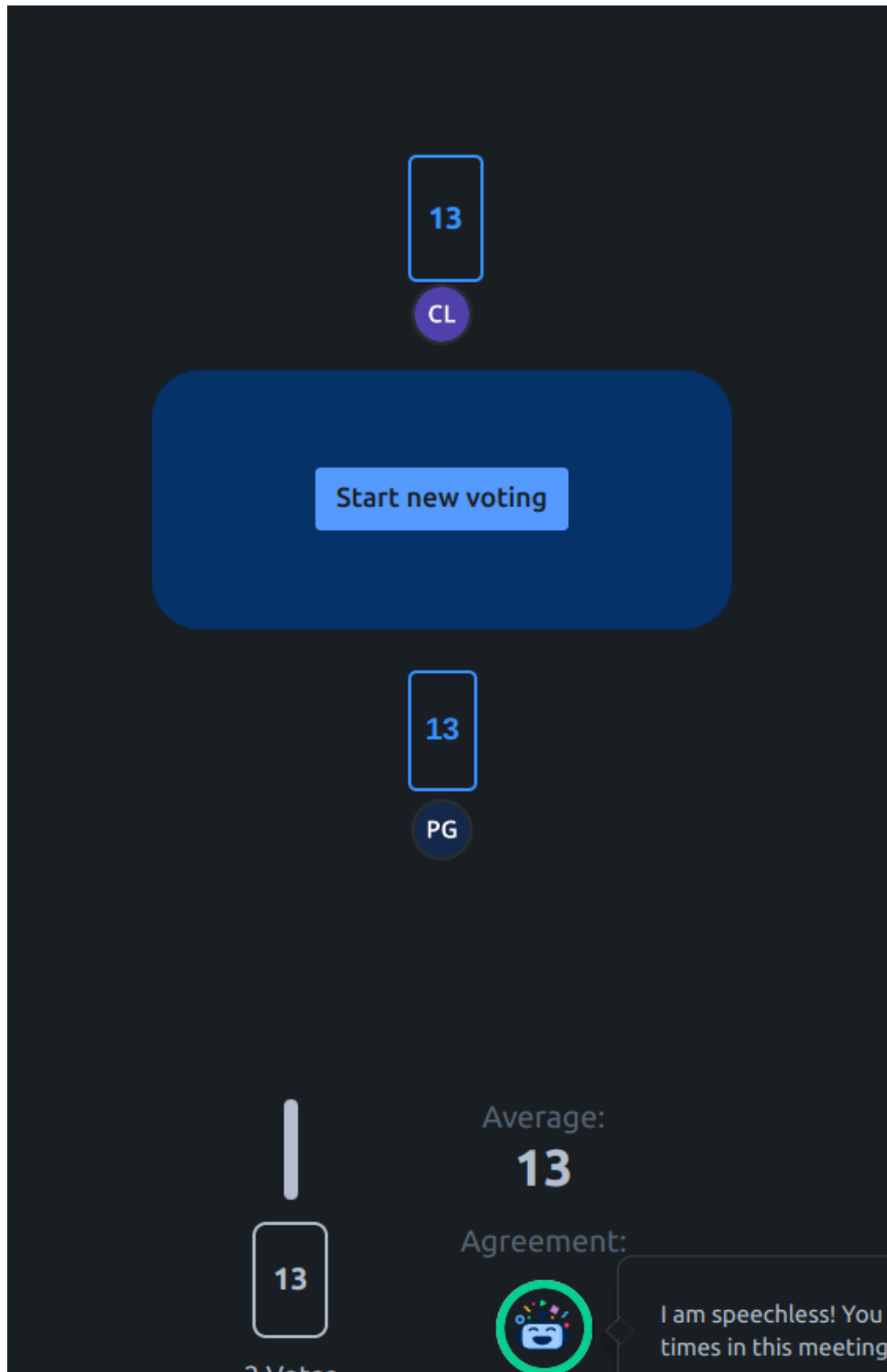
### 13. Creación de elementos dentro de la plataforma -> 21





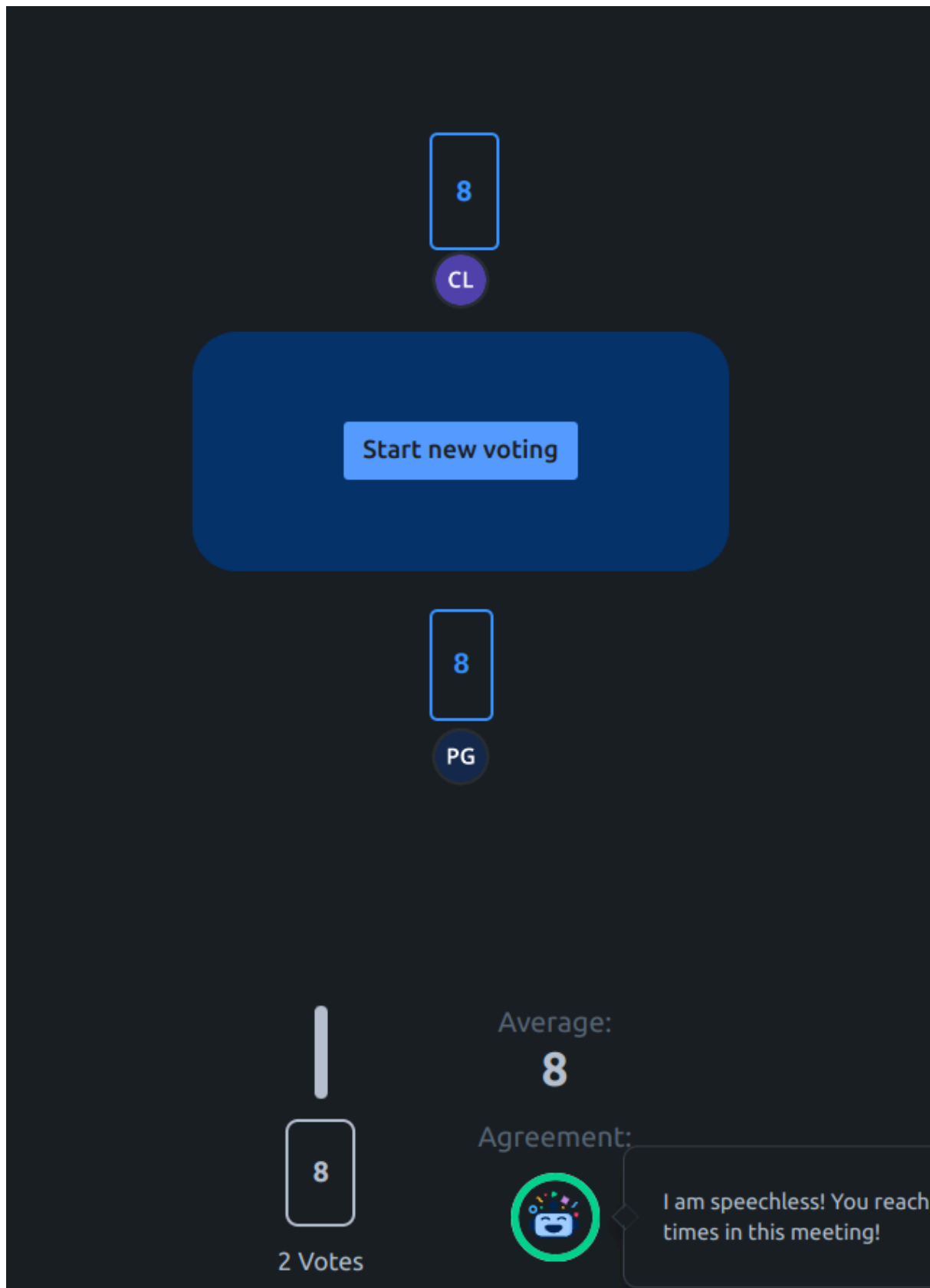


#### 14. Administración de elementos dentro de la plataforma -> 21





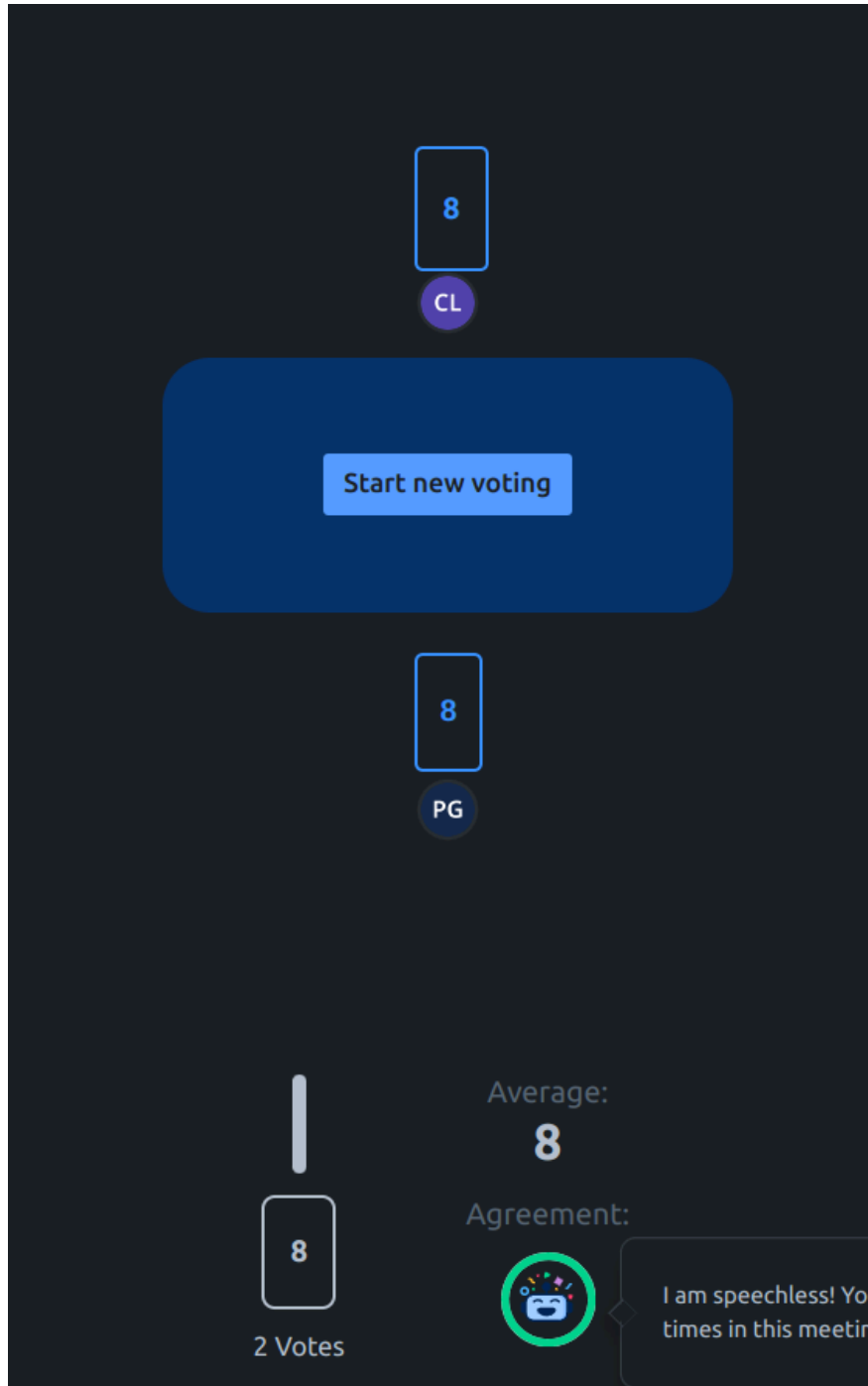
## 15. Generación de facturas -> 8





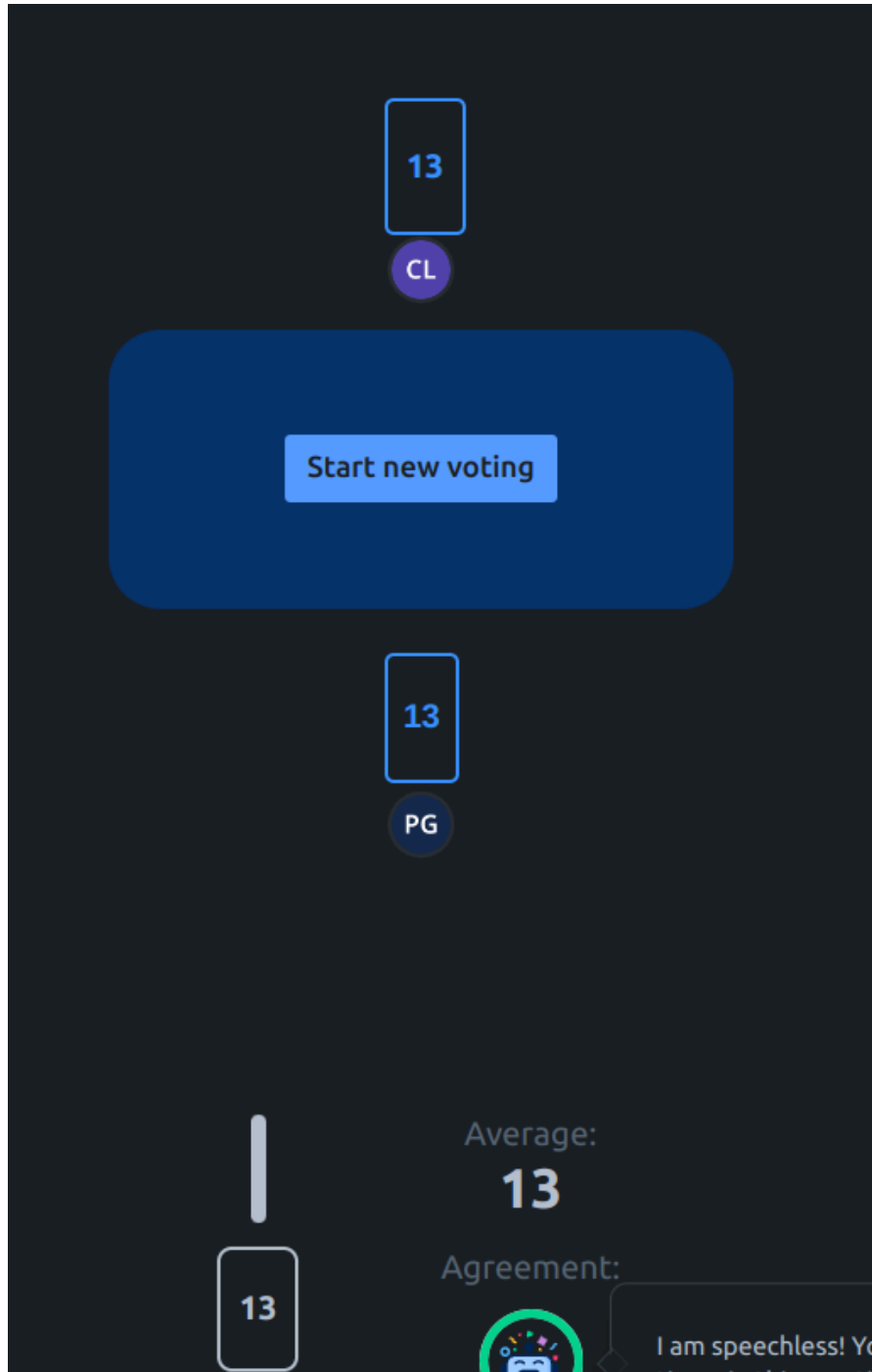


## 16. Generación de comprobantes -> 8



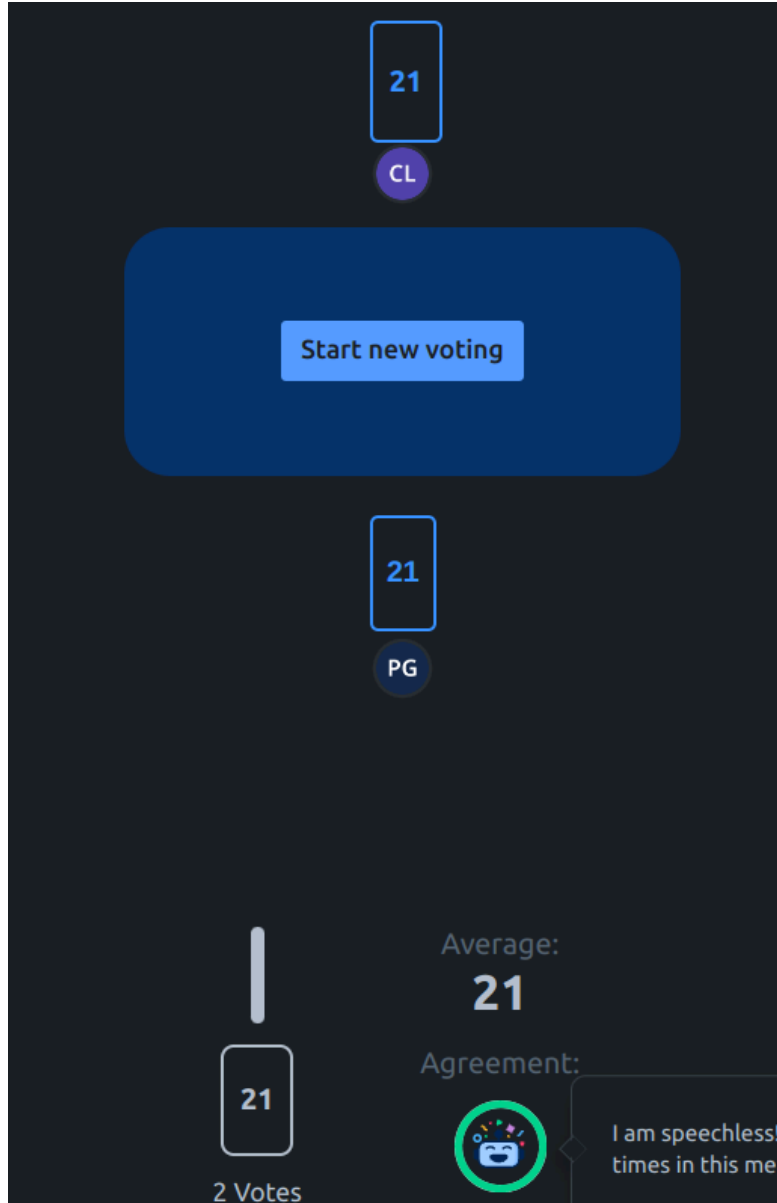


## 17. Creación de jornada laboral



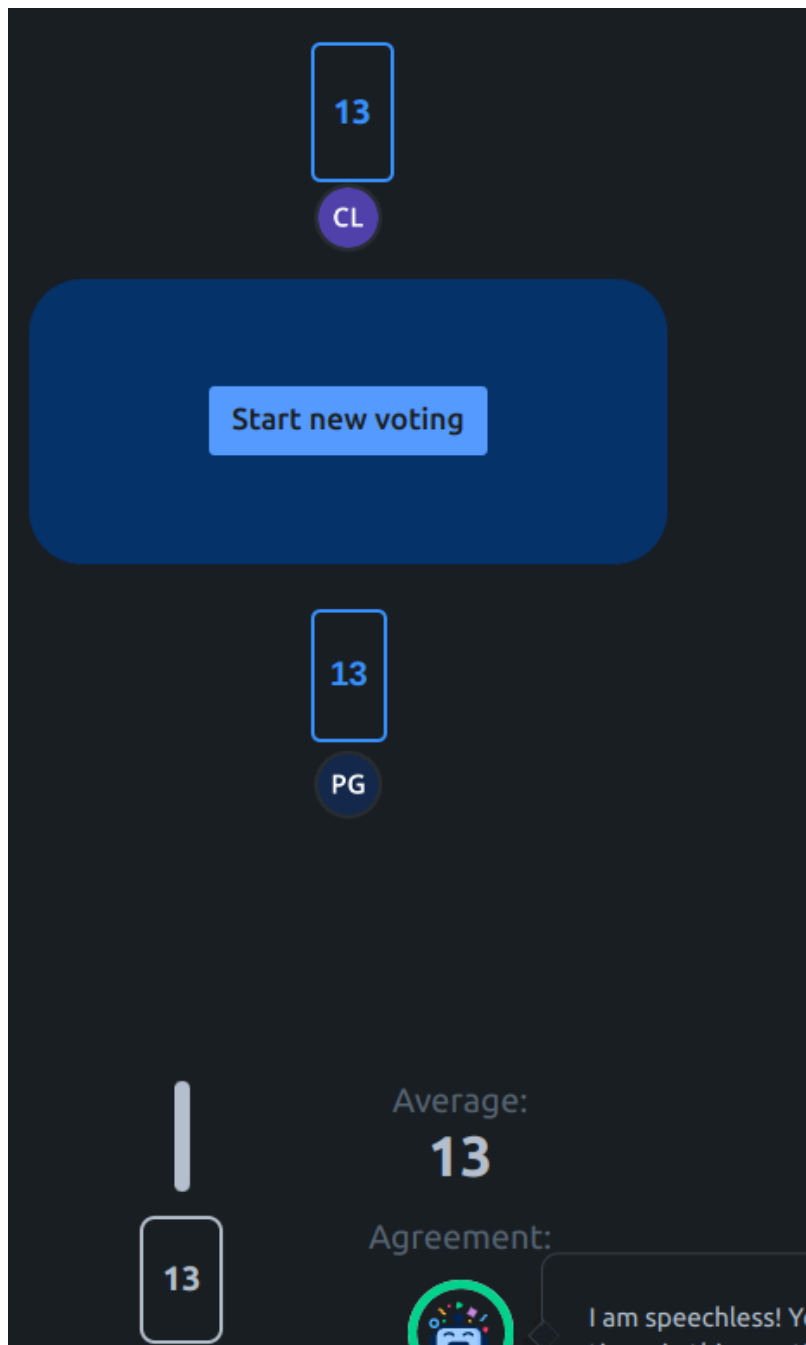


## 18. Edición de jornada laboral



## 19. Reprogramación de citas como cliente

Que los clientes puedan cambiar la fecha y hora de su cita en espacios disponibles



## 20. Cancelaciones de citas -> 21

Al reservar la cita preguntar si tiene el interes o posibilidad de poder llegar antes si en caso se cancela una cita en el horario que el puede. Al cancelarse la cita en un horario especifico, si hay alguien como comodin preguntarle si puede llegar, si es asi, se traslada la cita, de lo contrario se envia una notificacion a todos los de las siguientes citas indicando

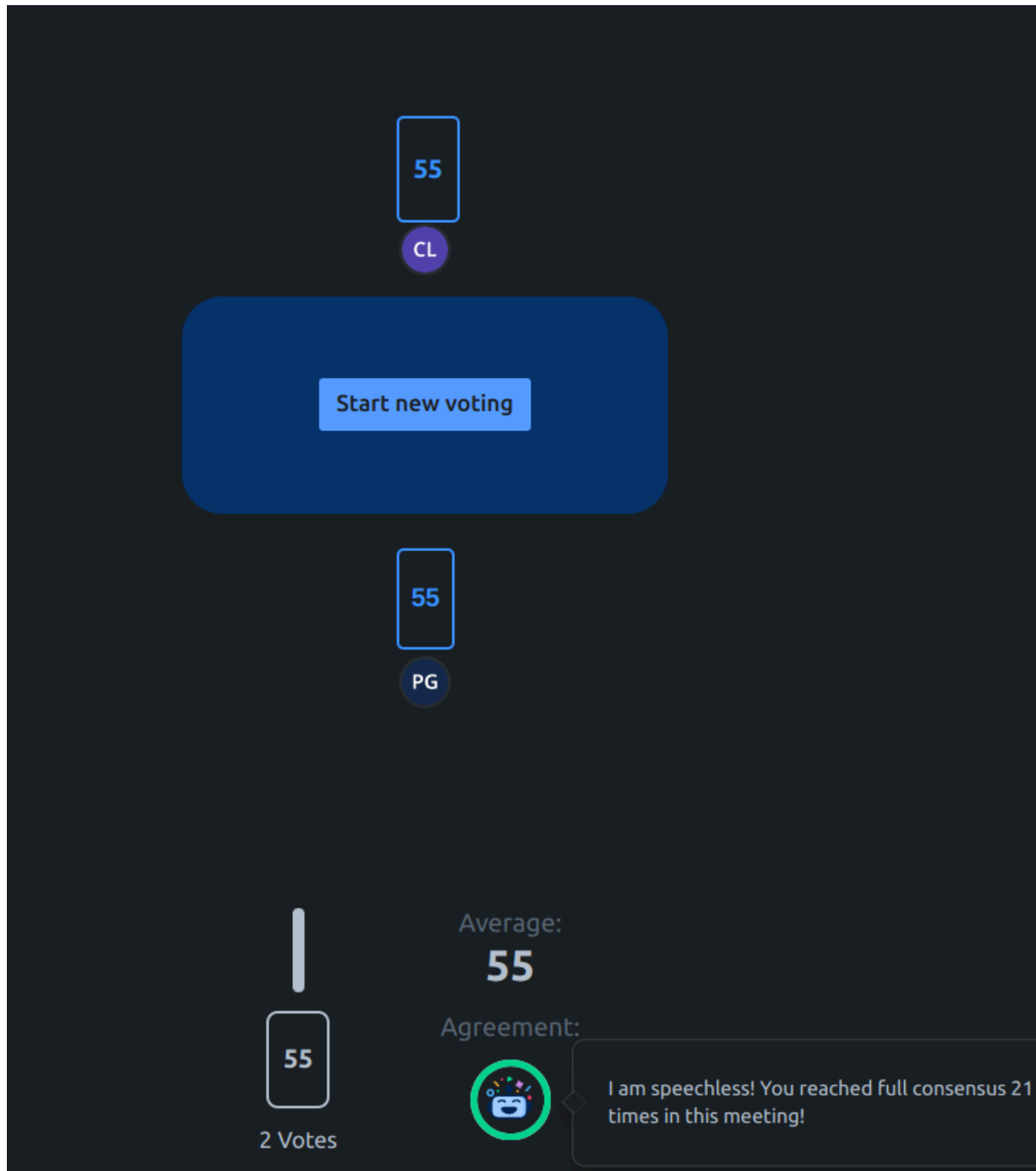


que hay espacio disponible y el primero que lo acepte se le cambia la cita y el espacio anterior queda disponible para los demás. Y si no se responde en 30 minutos dejar el espacio disponible para cualquier persona.

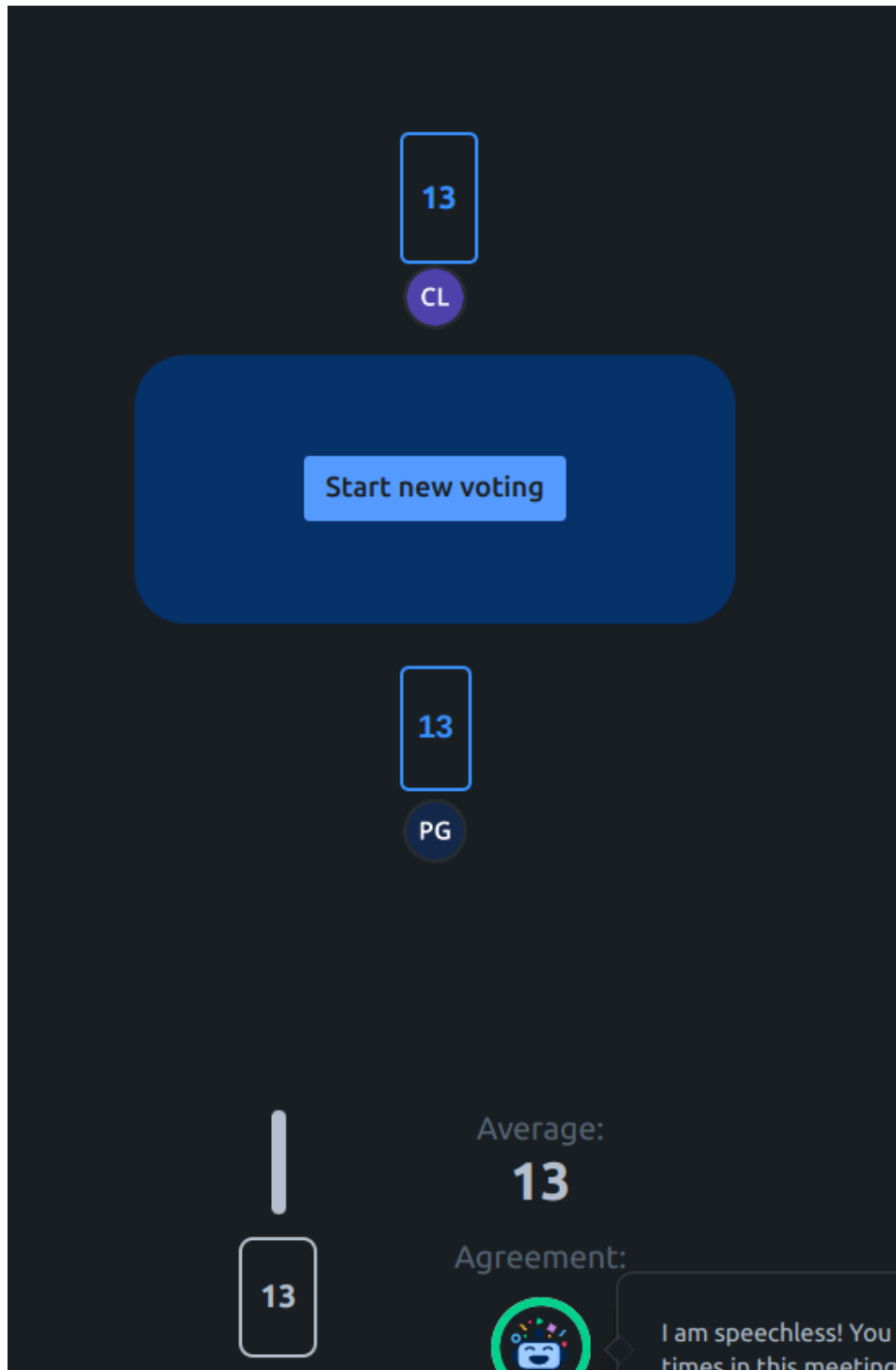
The screenshot displays a voting interface on a dark background. At the top, a blue box contains the number '21' above a purple circle with 'CL'. Below this is a large blue rounded rectangle with a white button labeled 'Start new voting'. Underneath the button, another blue box shows '21' above a dark blue circle with 'PG'. At the bottom left, a vertical bar is next to a box with '21' and the text '2 Votes' below it. To the right, the text 'Average: 21' is shown, followed by 'Agreement:'. Below 'Agreement:' is a green circular icon with a speech bubble and confetti. A speech bubble points to this icon, containing the text: 'I am speechless! You reached full consensus 20 times in this meeting!'.



21. Login (jwt y a2f) y mostrar las opciones según el rol del cliente o administrador -> 55

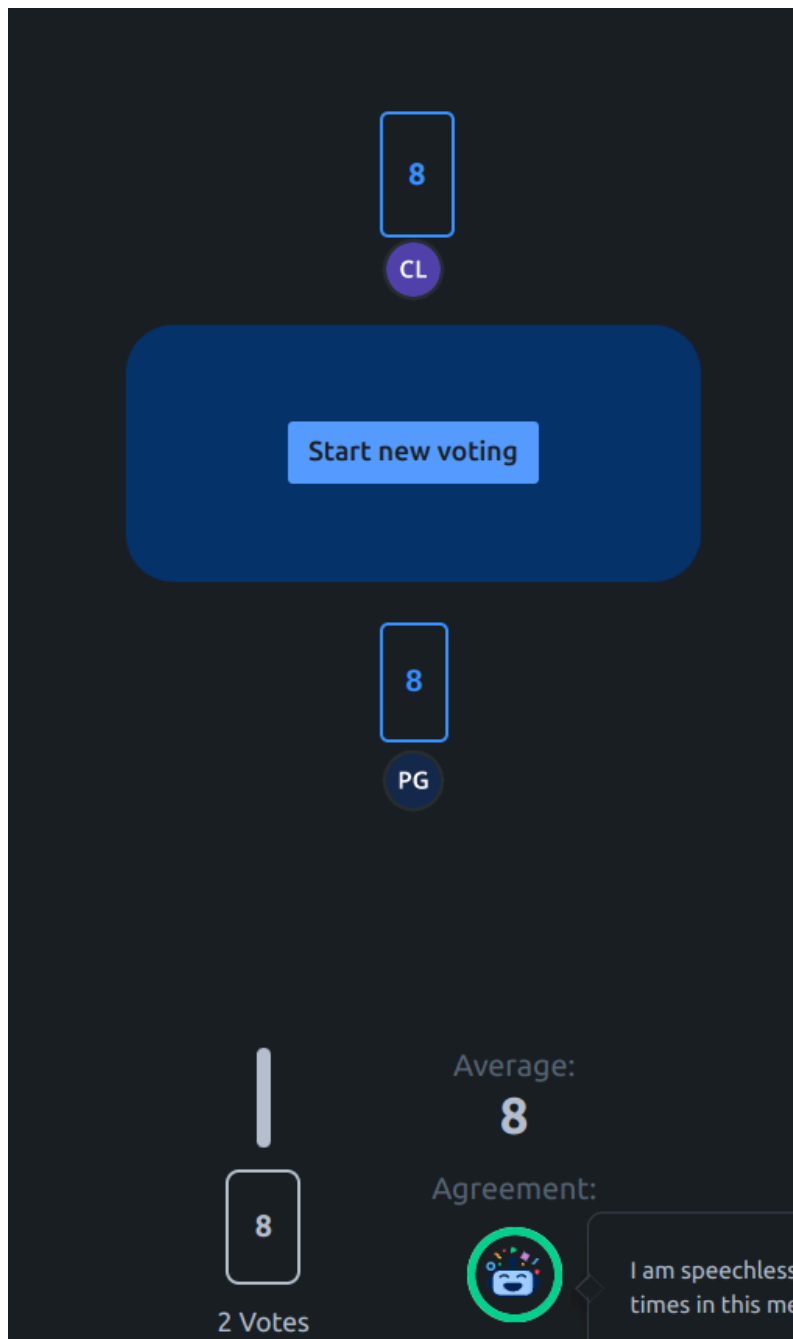


22. Reprogramación de citas como administrador-> 55





## 23. Creación y edición de categorías

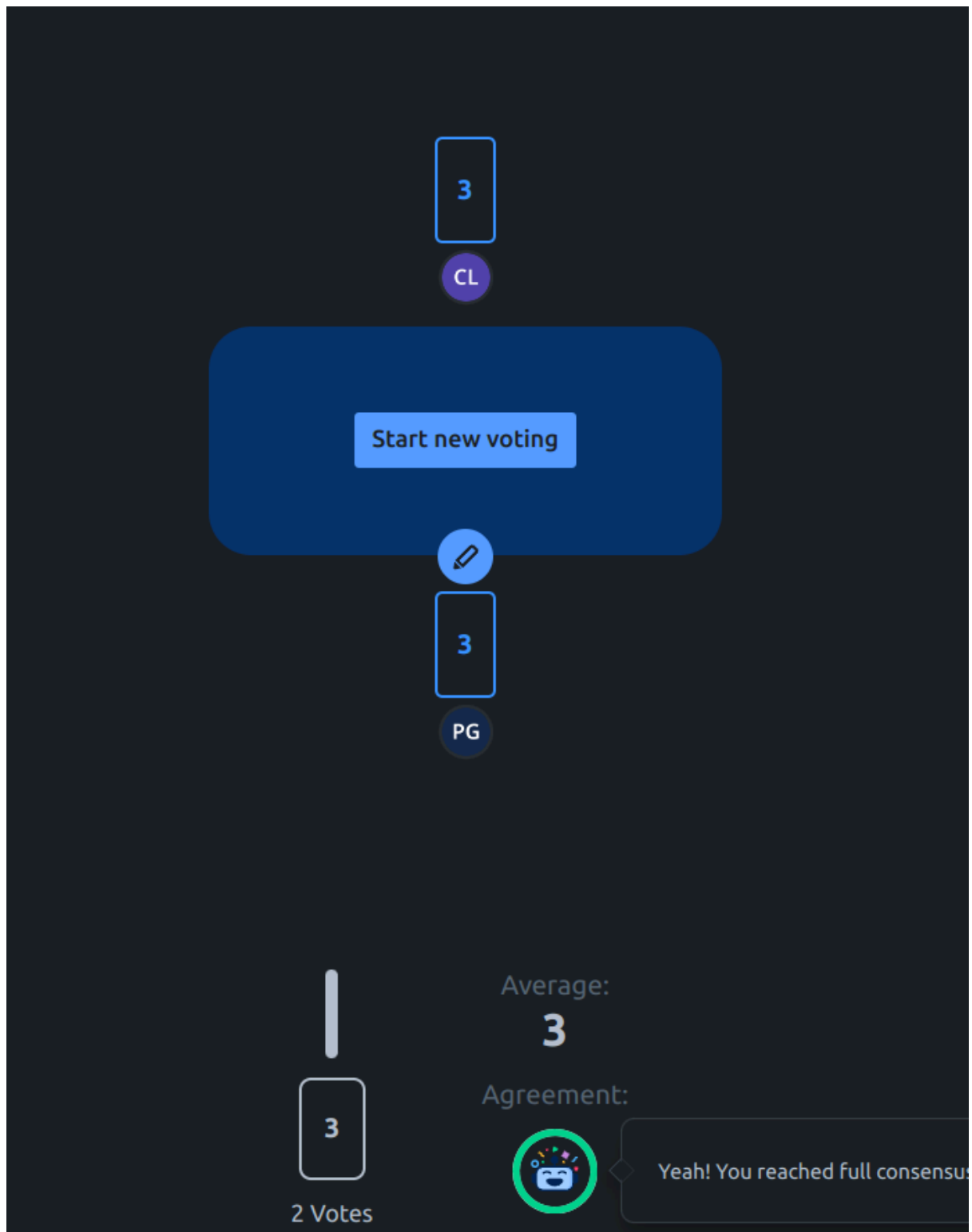






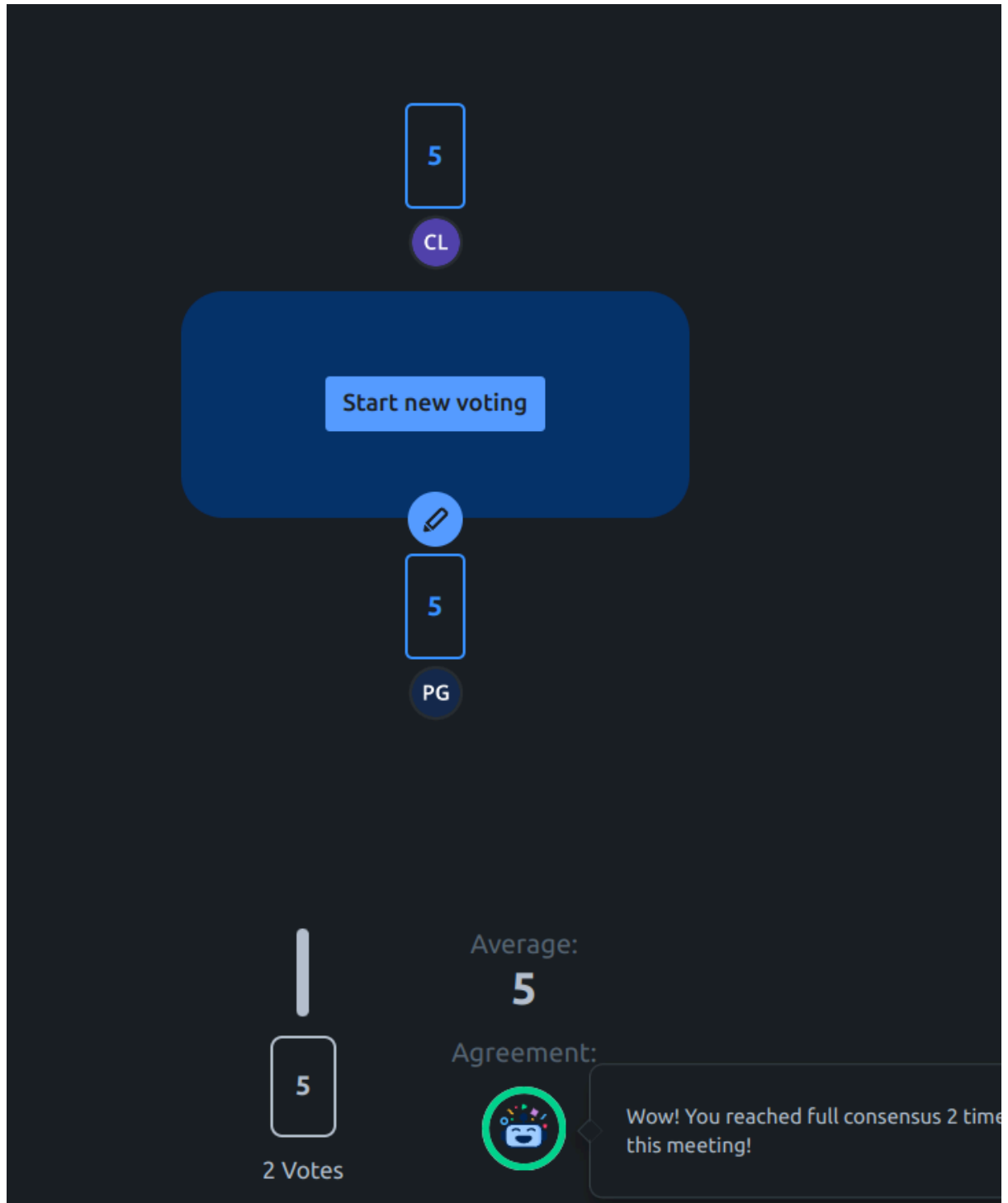
## **Cientes**

1. Registrarse en la plataforma -> 3



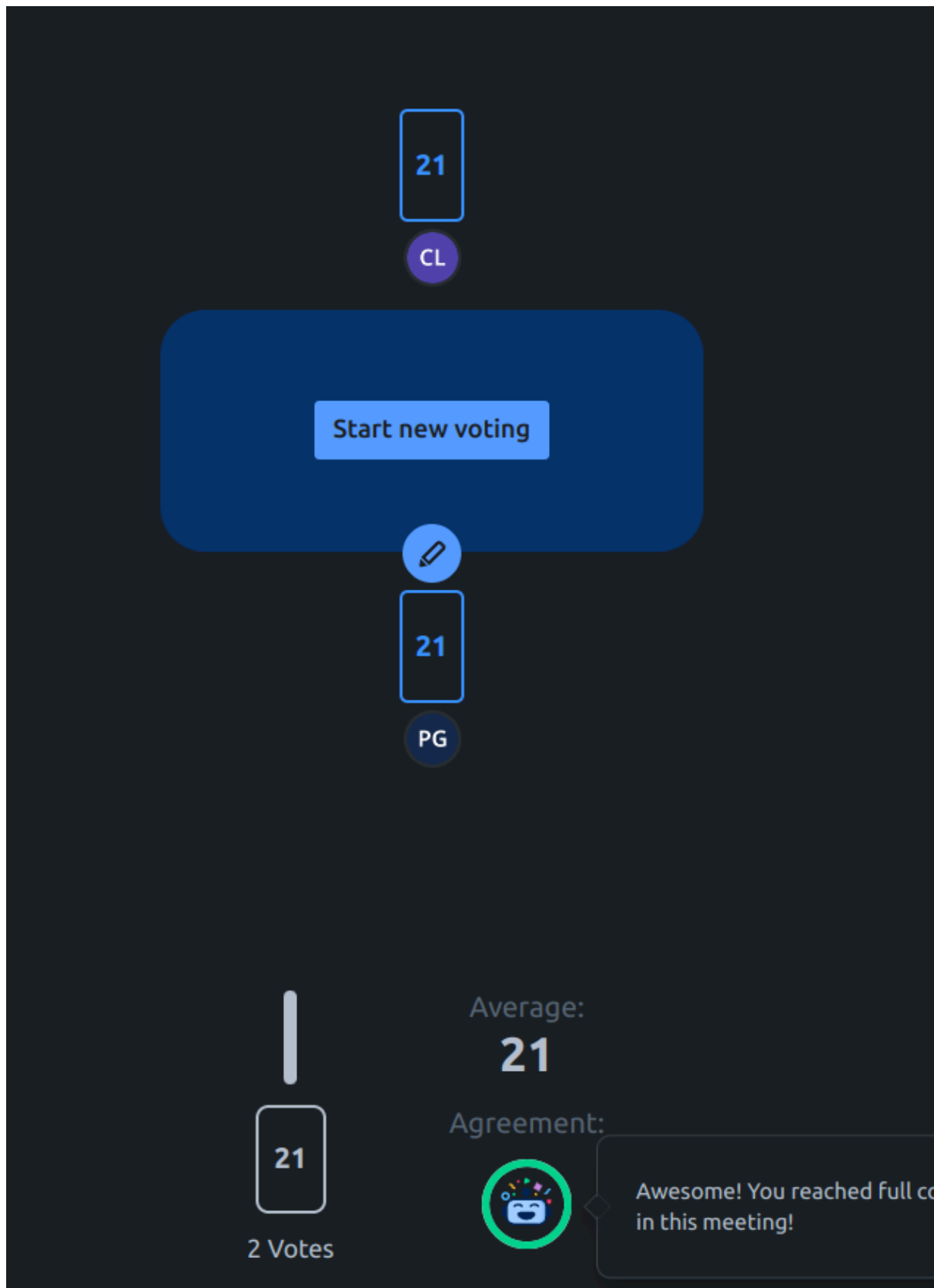


## 2. Recuperar contraseña -> 5



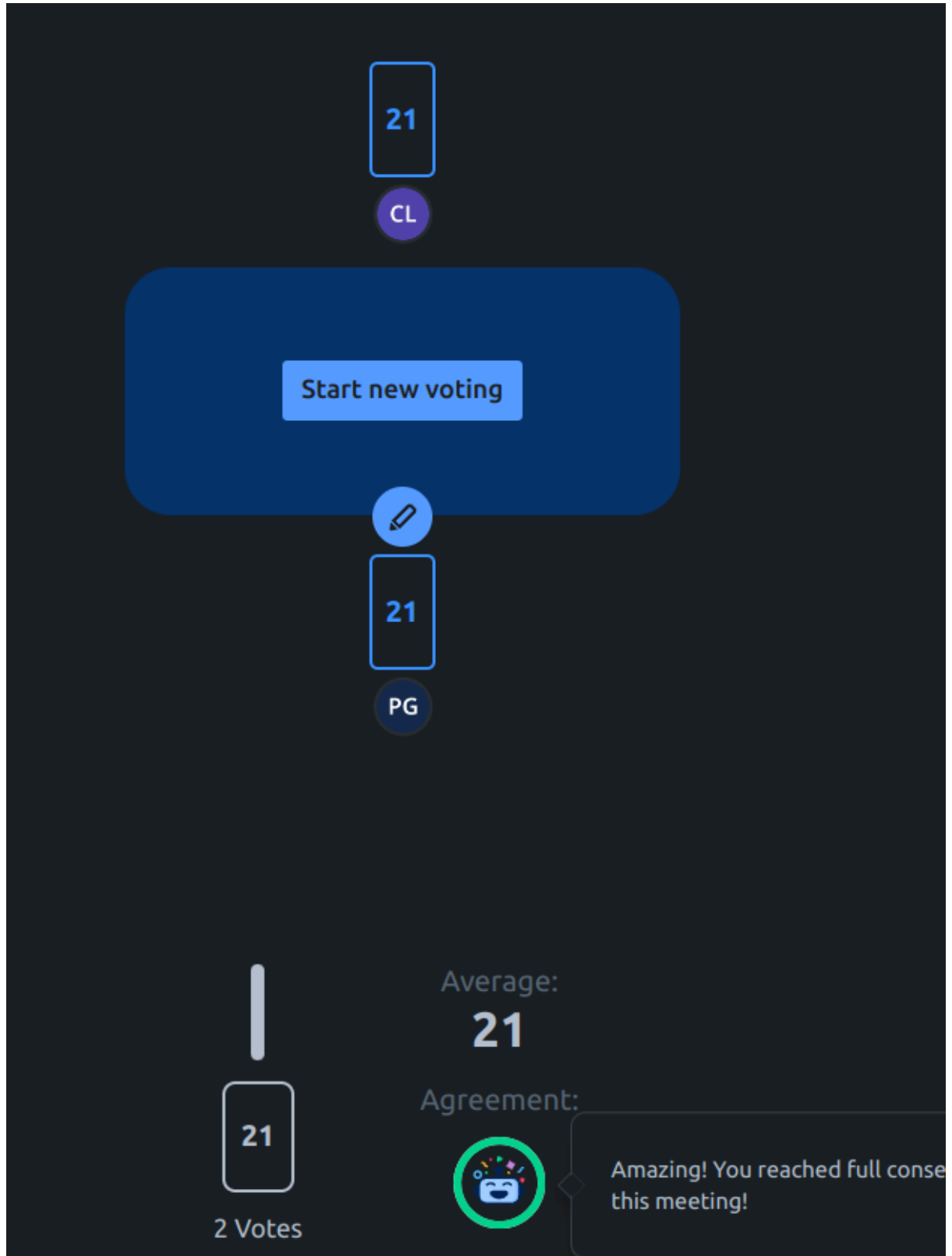


### 3. Visualización de calendario personal -> 21





#### 4. Agendar citas -> 21



Universidad de San Carlos de Guatemala  
Centro Universitario de Occidente  
División de Ingeniería  
Ingeniería en Ciencias y Sistemas  
Análisis y Diseño de Sistemas 1



**USAC**  
TRICENTENARIA  
Universidad de San Carlos de Guatemala

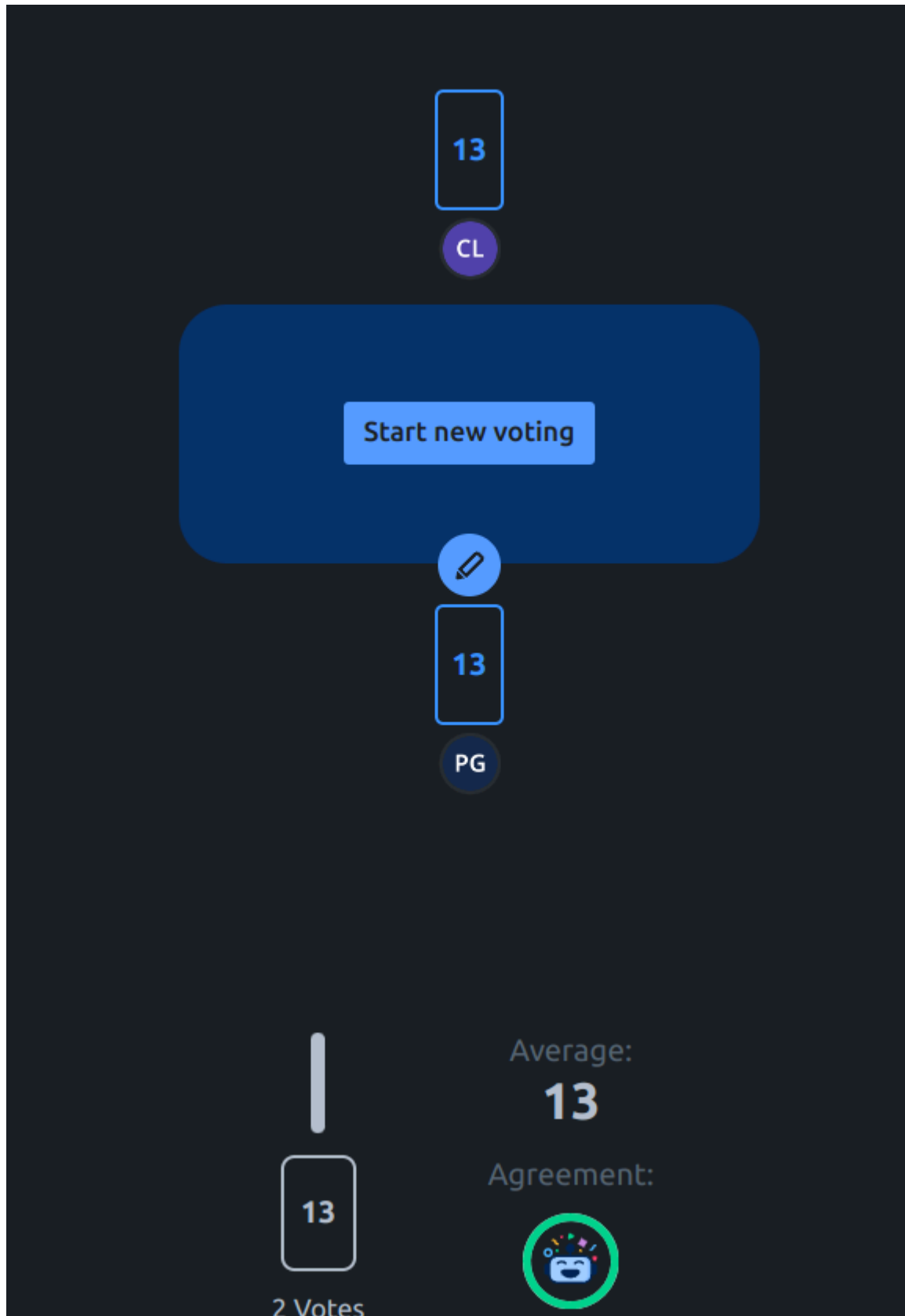
Universidad de San Carlos de Guatemala  
Centro Universitario de Occidente  
División de Ingeniería  
Ingeniería en Ciencias y Sistemas  
Análisis y Diseño de Sistemas 1



**USAC**  
TRICENTENARIA  
Universidad de San Carlos de Guatemala

5. Cancelar cita -> 13





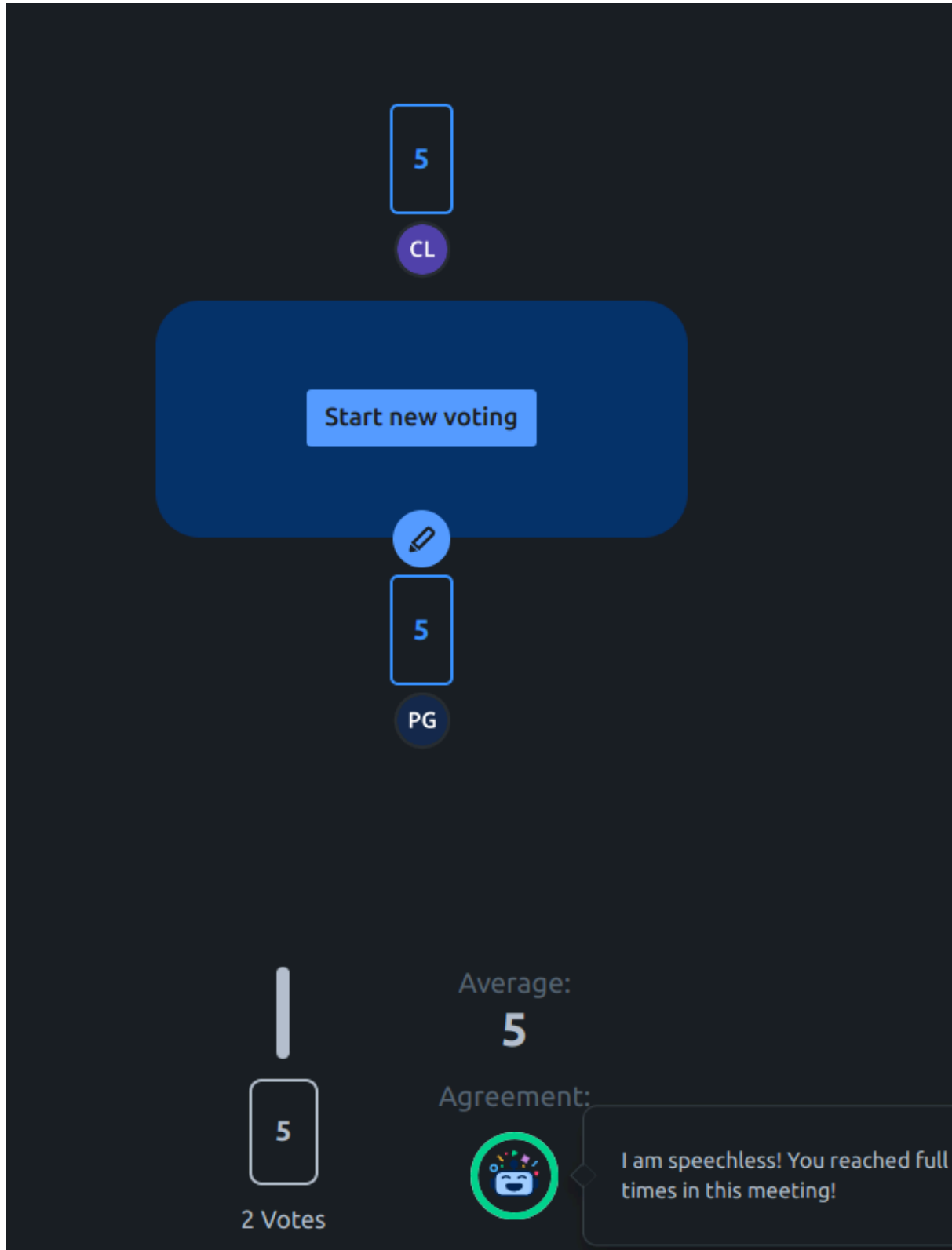
Universidad de San Carlos de Guatemala  
Centro Universitario de Occidente  
División de Ingeniería  
Ingeniería en Ciencias y Sistemas  
Análisis y Diseño de Sistemas 1



**USAC**  
TRICENTENARIA  
Universidad de San Carlos de Guatemala

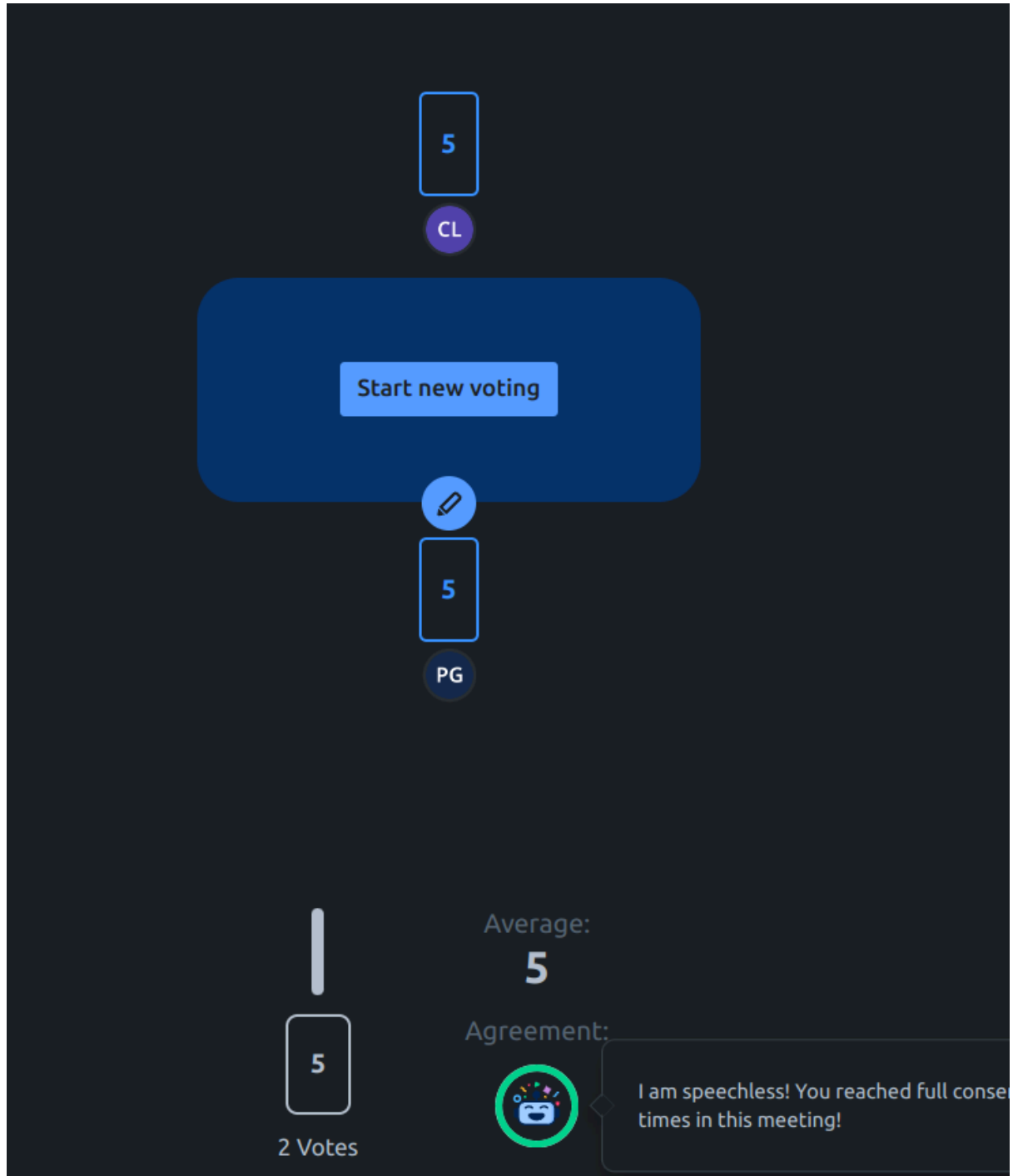


## 6. Visualización del historial de citas



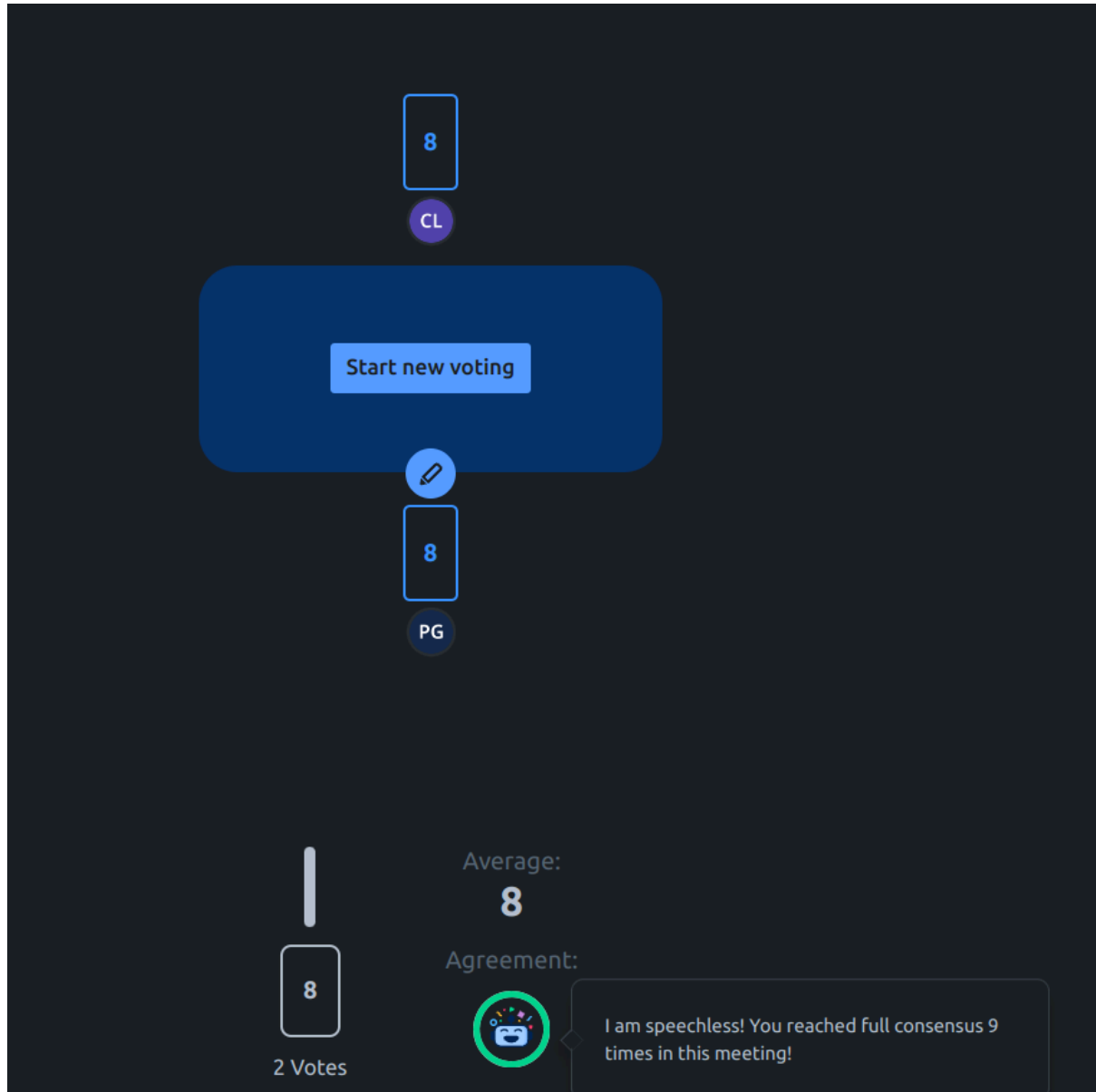


## 7. Mensajes de notificaciones





## 8. Ver comprobantes y facturas



## Diagramas

### Casos de Uso

#### Registrar Usuario CU01

- **Descripción:** Permite al usuario registrarse en el Sitio



- **Actores Involucrados:** Usuario
- **Flujo Principal:**
  - El usuario accede al formulario de registro.
  - El usuario completa la información requerida (nombre, apellido, email, contraseña, dirección, nit).
  - El sistema valida la información.
  - El sistema crea la cuenta para el usuario y lo redirige a la página de inicio.

#### Iniciar Sesión (Autenticación 2FA) CU02

- **Descripción:** Permite al usuario iniciar sesión en el sistema con autenticación de dos factores (2FA).
- **Actores involucrados:** Usuario, Comprador.
- **Flujo Principal:**
  - El usuario accede al formulario de inicio de sesión.
  - El usuario ingresa sus credenciales (correo electrónico y contraseña).
  - El sistema valida las credenciales.
  - El sistema solicita un código de verificación adicional (enviado por SMS o correo electrónico).
  - El usuario ingresa el código de verificación.
  - El sistema verifica el código 2FA.
  - El sistema autentica al usuario y lo redirige a su área de página principal.

#### Recuperar Contraseña CU03

- **Descripción:** Permite al usuario recuperar su contraseña en caso de olvido.
- **Actores involucrados:** Usuario.
- **Flujo Principal:**
  - El usuario selecciona la opción de "Recuperar contraseña".
  - El usuario ingresa su correo electrónico registrado.
  - El sistema envía un enlace de recuperación de contraseña al correo electrónico.
  - El usuario accede al enlace y restablecer su contraseña.
  - El sistema valida y confirma el restablecimiento de la contraseña.
  - El usuario puede iniciar sesión con la nueva contraseña.

#### Buscar Productos CU04

- **Descripción:** Permite al usuario buscar productos en la tienda en línea.
- **Actores involucrados:** Usuario, Comprador.
- **Flujo Principal:**
  - El usuario accede a la página principal.
  - El usuario utiliza la barra de búsqueda o los filtros de categoría para encontrar productos.



- El sistema muestra los resultados de la búsqueda.
- El usuario selecciona un producto para ver más detalles.

#### Agregar Productos al Carrito CU05

- **Descripción:** Permite al usuario agregar productos a su carrito de compras.
- **Actores involucrados:** Usuario, Comprador.
- **Flujo Principal:**
  - El usuario navega por la tienda y selecciona un producto.
  - El usuario ingresa la cantidad deseada.
  - El sistema agrega el producto al carrito.
  - El usuario continúa navegando o accede al carrito de compras.

#### Realizar Compra CU06

- **Descripción:** Permite al usuario realizar una compra de los productos agregados al carrito.
- **Actores involucrados:** Comprador.
- **Flujo Principal:**
  - El usuario accede al carrito de compras.
  - El usuario verifica los productos seleccionados, las cantidades y el total a pagar.
  - El usuario elige el método de pago y la opción de entrega (en tienda o domicilio).
  - El sistema procesa el pago y confirma la compra.
  - El usuario recibe una confirmación por correo electrónico.

#### Gestión de Inventario CU07

- **Descripción:** Permite a los administradores y ayudantes gestionar el inventario de la tienda.
- **Actores involucrados:** Administrador, Ayudante.
- **Flujo Principal:**
  - El administrador o ayudante accede a la sección de gestión de productos.
  - El sistema muestra el inventario actual.
  - El administrador o ayudante selecciona un producto para actualizar la cantidad o modificar detalles.
  - El sistema guarda los cambios y actualiza el inventario.

#### Generar Reportes de Ventas CU08

- **Descripción:** Permite al administrador generar reportes de ventas.



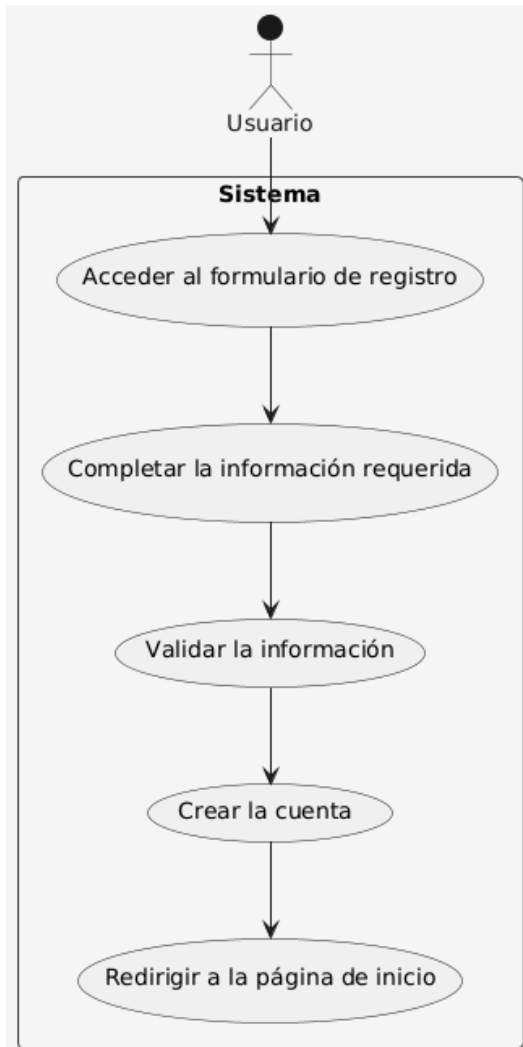
- **Actores involucrados:** Administrador.
- **Flujo Principal:**
  - El administrador accede al dashboard de administración.
  - El administrador selecciona la opción para generar reportes.
  - El sistema presenta opciones de filtros para los reportes (fecha, categoría de productos, etc.).
  - El administrador genera el reporte.
  - El sistema exporta el reporte en el formato solicitado





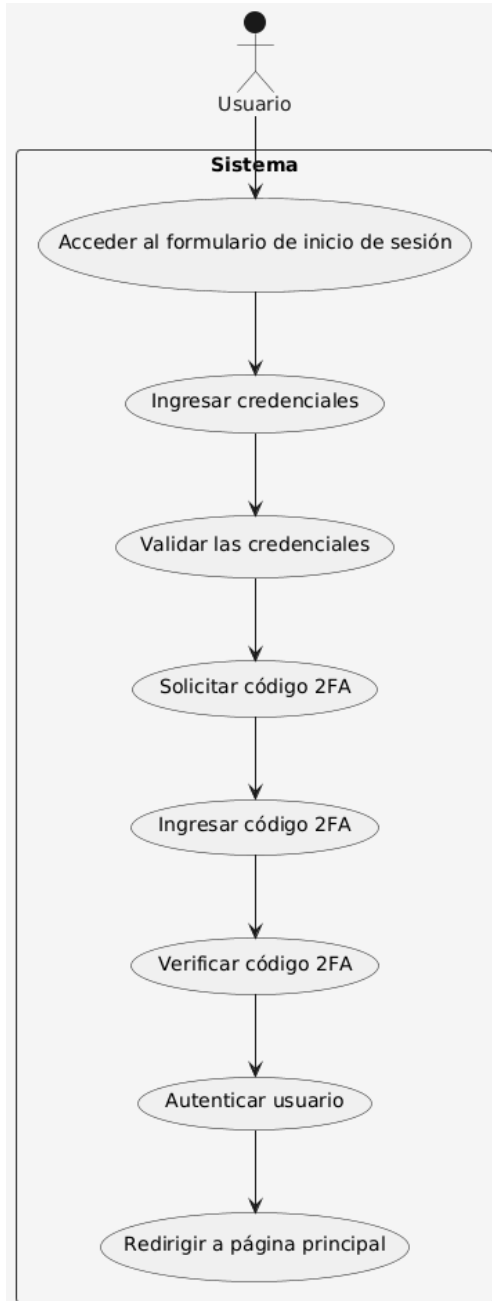
## Diagramas de Casos de Uso

CU01



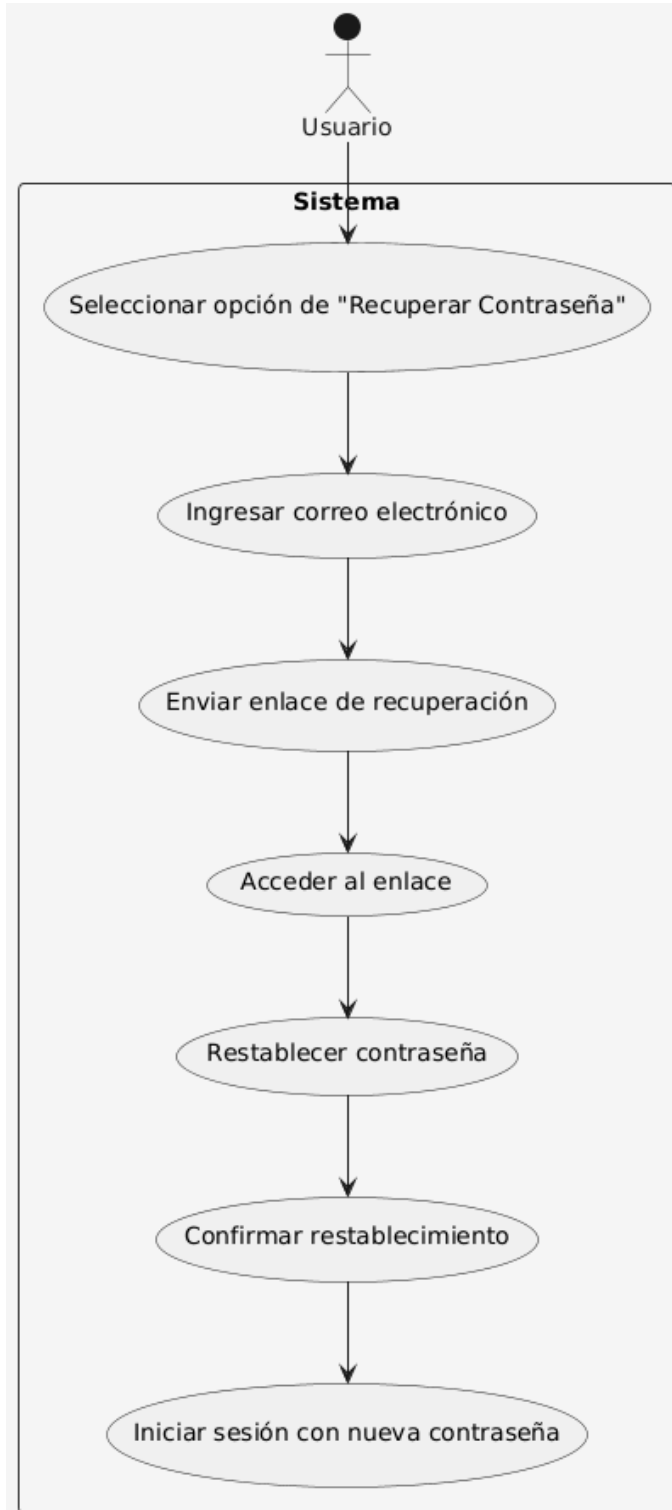


CU02



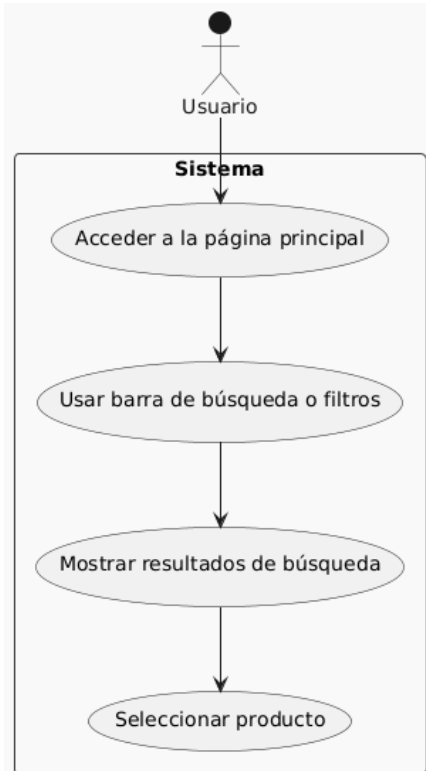


CU03



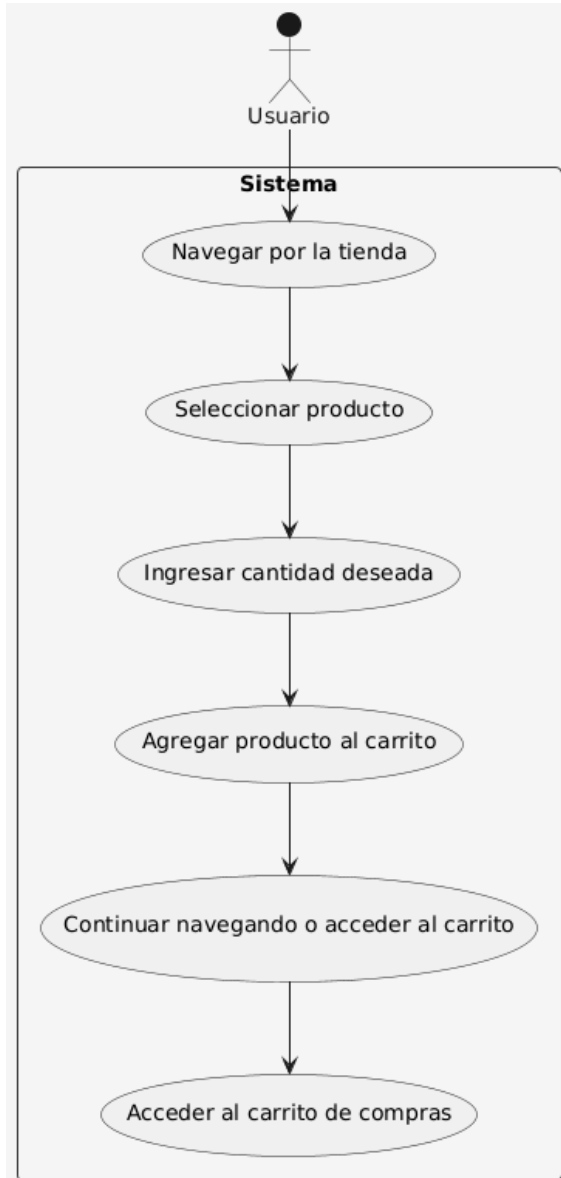


CU04



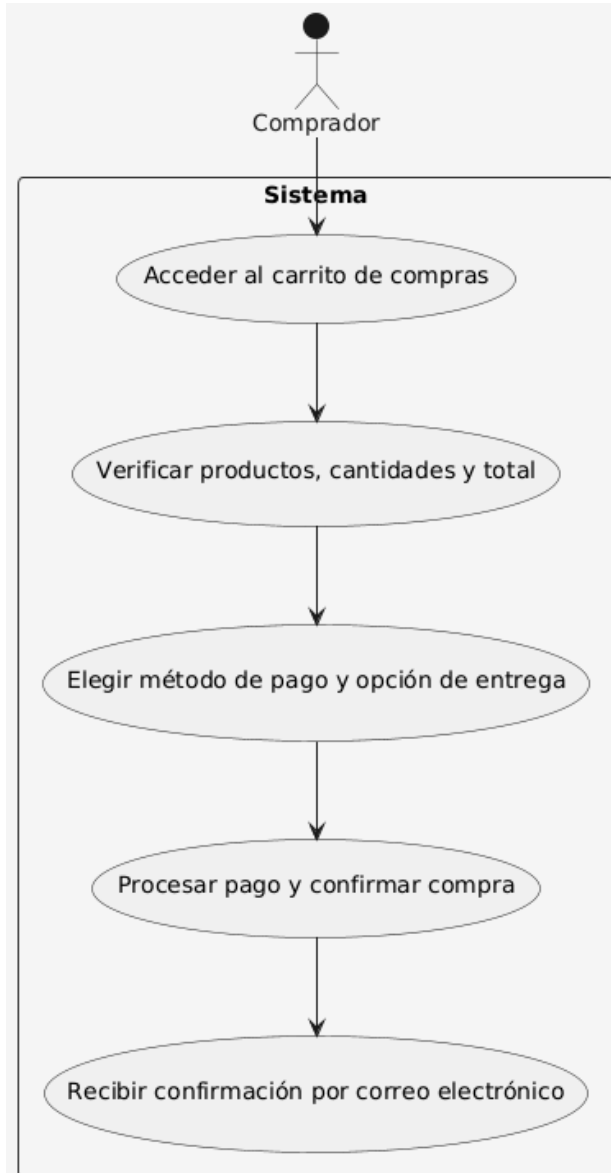


CU05



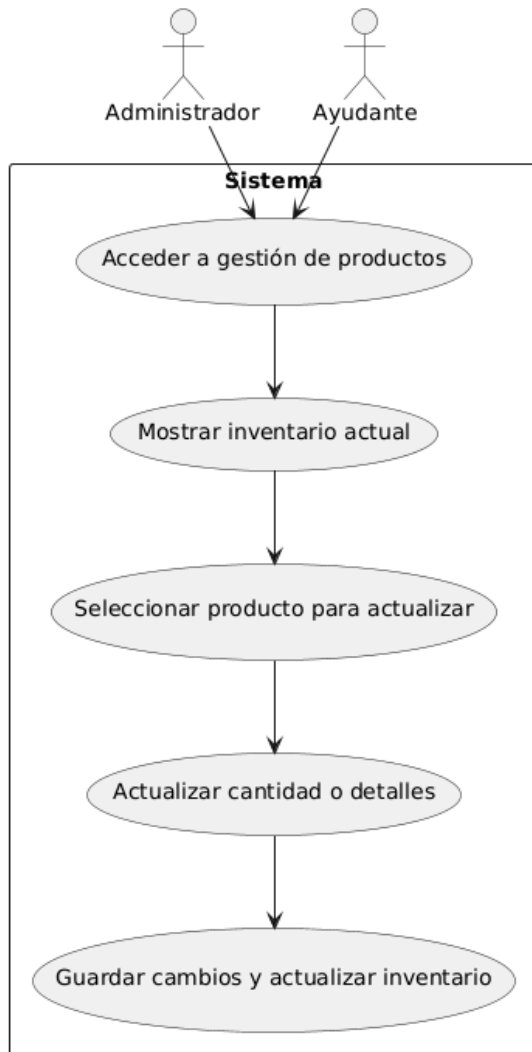


CU06



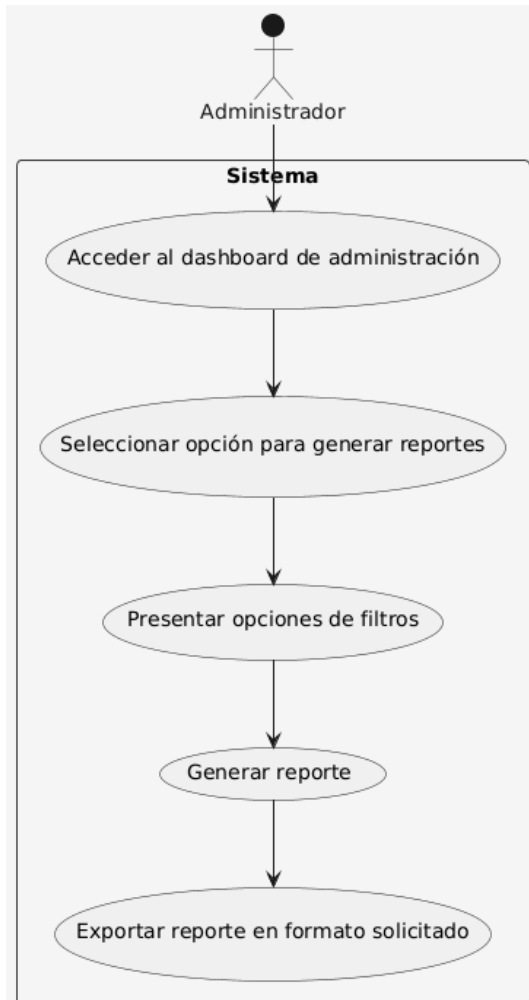


CU07





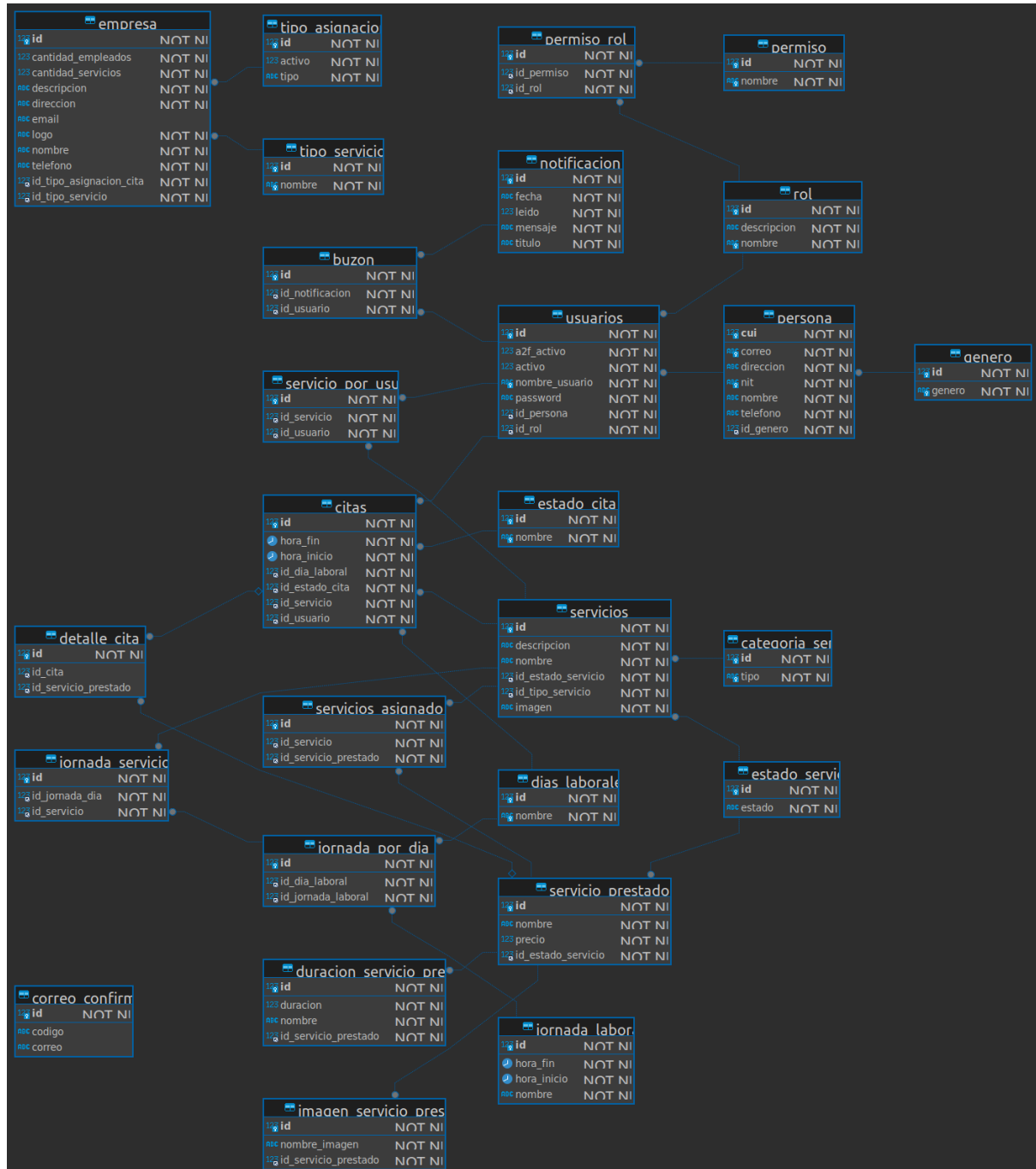
CU08





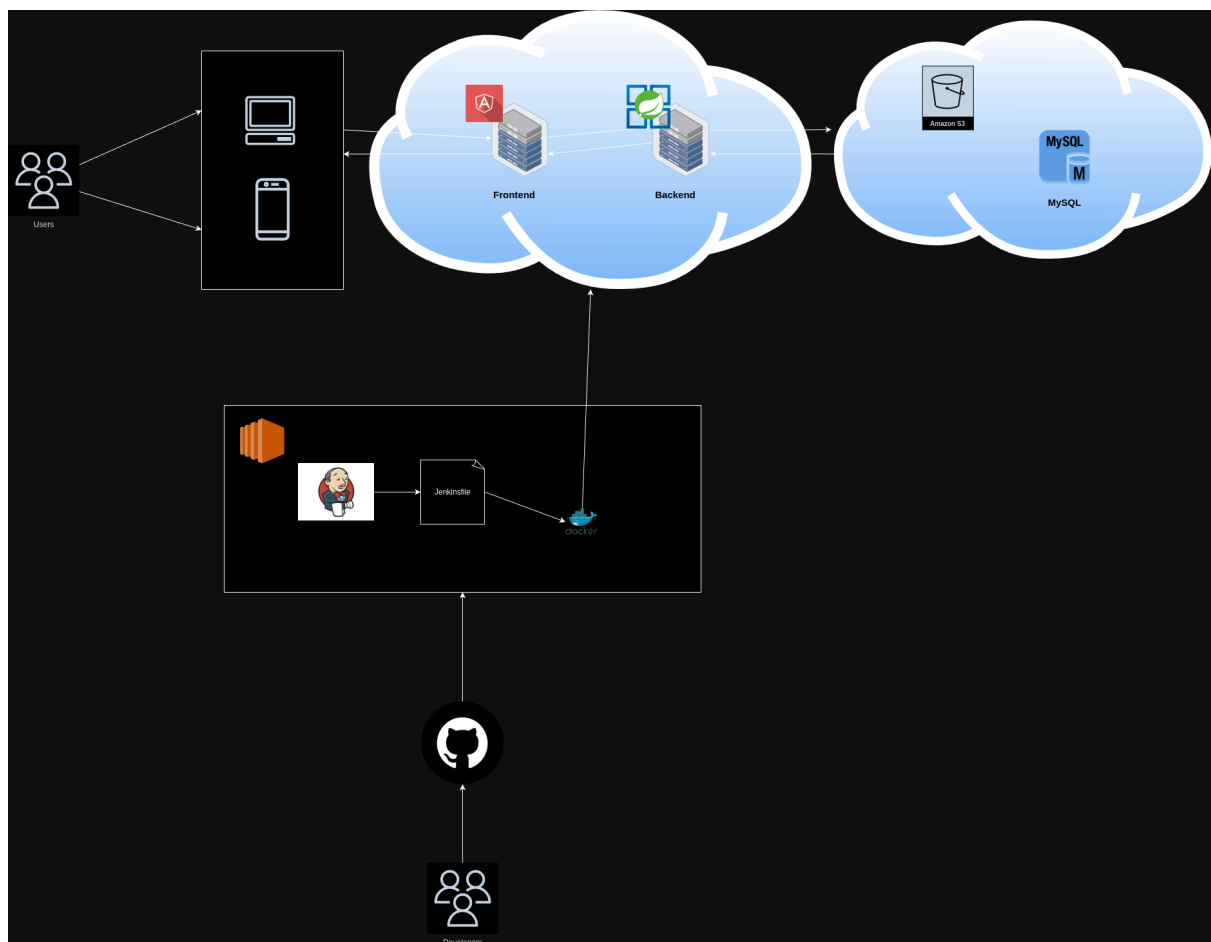


## Diagrama de tablas



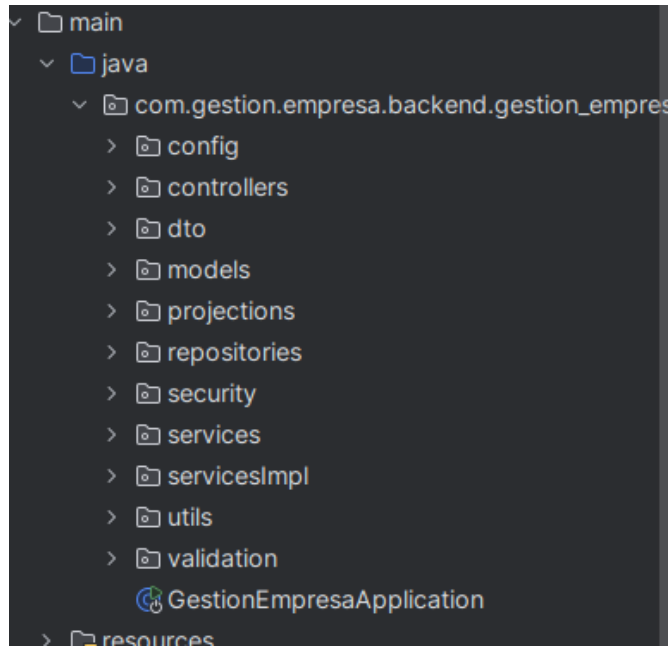


## Diagrama de arquitectura

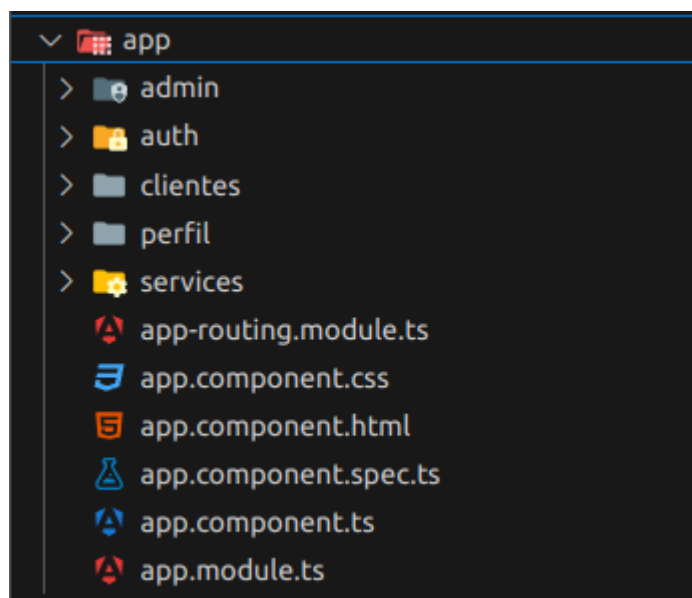




## Backend



## Frontend





## Requerimientos Técnicos

- **Procesador:** Intel Core i5 de 10ª generación o superior.
- **Memoria RAM:** 8 GB.
- **Espacio de almacenamiento:** 50 GB.

## Sistema Operativo

**Requisitos Adicionales:** Deben estar instaladas las siguientes herramientas:

### Spring Boot

- **Descripción:** Spring Boot es un potente framework para el desarrollo de aplicaciones Java y microservicios. Proporciona una forma simplificada de crear aplicaciones con configuraciones automáticas y un enfoque en la rapidez de desarrollo.
- **Versión recomendada:** 3.x o superior.
- **Instalación:** No requiere instalación específica. Para iniciar un nuevo proyecto, puedes usar [Spring Initializr](#) para generar la estructura básica del proyecto. Este generador te permite seleccionar las dependencias necesarias, configuraciones de proyecto, y el tipo de packaging.

### MySQL

- **Descripción:** MySQL es un sistema de gestión de bases de datos relacional muy utilizado, conocido por su rapidez y facilidad de uso. Ideal para gestionar datos como usuarios, productos y pedidos.
- **Versión recomendada:** 8.x o superior.
- **Instalación en Windows:**
  1. Descarga el instalador desde [mysql.com](https://mysql.com).
  2. Durante la instalación, selecciona el tipo de instalación (típica o personalizada) según tus necesidades.
  3. Asegúrate de anotar la contraseña de root que establezcas, ya que la necesitarás para conectarte a la base de datos.

### Instalación en Linux:

bash

Copiar código

```
sudo apt update
```

```
sudo apt install mysql-server
```

Después de la instalación, puedes asegurar la instalación de MySQL ejecutando:

bash

Copiar código

```
sudo mysql_secure_installation
```



- Esto te guiará a través de varias configuraciones para mejorar la seguridad de tu servidor MySQL.

## Angular

- **Descripción:** Angular es un poderoso framework para el desarrollo de frontend en aplicaciones web. Facilita la creación de aplicaciones web dinámicas y responsivas.
- **Versión recomendada:** 13.x o superior.

**Instalación en Windows y Linux:** Necesitas tener Node.js instalado. Luego, ejecuta:

bash

Copiar código

```
npm install -g @angular/cli
```

- Esto instalará la interfaz de línea de comandos de Angular, permitiéndote crear y gestionar proyectos de Angular con facilidad.

## JWT (JSON Web Tokens)

- **Descripción:** Los JSON Web Tokens son un método seguro de autenticación basado en tokens, que permite verificar la identidad de los usuarios sin la necesidad de mantener el estado en el servidor.
- **Versión recomendada:** 8.x o superior.

**Instalación:** No requiere instalación, se implementa como una librería en el proyecto:

bash

Copiar código

```
npm install jsonwebtoken
```

- 

## 2FA (Autenticación de Dos Factores)

- **Descripción:** Proporciona una capa adicional de seguridad, requiriendo no solo la contraseña, sino también un segundo factor, como un código enviado al teléfono.

**Instalación:** Ejecuta el siguiente comando para instalar las librerías necesarias:

bash

Copiar código

```
npm install speakeasy qrcode
```

- 

## Tailwind CSS



- **Descripción:** Tailwind CSS es un framework CSS utilitario que facilita la creación de interfaces de usuario modernas y responsivas sin necesidad de escribir CSS personalizado desde cero.
- **Versión recomendada:** 3.x o superior.

#### Instalación en Windows y Linux:

bash

Copiar código

```
npm install -D tailwindcss
```

```
npm run tailwindcss init
```

- Esto crea un archivo de configuración básico para Tailwind CSS en tu proyecto, permitiéndote personalizarlo según tus necesidades.

#### Flowbite

- **Descripción:** Flowbite es una librería de componentes UI que se basa en Tailwind CSS, ofreciendo una colección de componentes predefinidos que puedes utilizar para acelerar el desarrollo de tu interfaz de usuario.

#### Instalación en Windows y Linux:

bash

Copiar código

```
npm install flowbite
```

- 

#### Swagger

- **Descripción:** Swagger es una herramienta útil para documentar y probar APIs REST, permitiendo a los desarrolladores interactuar con la API de manera sencilla y clara.

#### Instalación en Windows y Linux: Se usa como dependencia en el backend:

bash

Copiar código

```
npm install swagger-ui-express
```

- 

#### Git

- **Descripción:** Git es un sistema de control de versiones que permite gestionar y rastrear cambios en el código fuente a lo largo del tiempo, facilitando la colaboración entre múltiples desarrolladores.



- **Instalación en Windows:** Descarga desde [git-scm.com](https://git-scm.com).

#### Instalación en Linux:

bash

Copiar código

```
sudo apt update
```

```
sudo apt install git
```

●

#### Postman

- **Descripción:** Postman es una plataforma popular utilizada por desarrolladores para probar y depurar APIs, proporcionando una interfaz intuitiva para enviar solicitudes y recibir respuestas.
- **Instalación en Windows:** Visita el sitio web oficial de [Postman](https://www.postman.com) y descarga el instalador para Windows (archivo .exe).

#### Instalación en Linux:

bash

Copiar código

```
sudo snap install postman
```

## Diagramas

### Casos de Uso

#### Registrar Usuario CU01

- **Descripción:** Permite al usuario registrarse en el Sitio
- **Actores Involucrados:** Usuario
- **Flujo Principal:**
  - El usuario accede al formulario de registro.
  - El usuario completa la información requerida (nombre, apellido, email, contraseña, dirección, nit).
  - El sistema valida la información.
  - El sistema crea la cuenta para el usuario y lo redirige a la página de inicio.

#### Iniciar Sesión (Autenticación 2FA) CU02

- **Descripción:** Permite al usuario iniciar sesión en el sistema con autenticación de dos factores (2FA).
- **Actores involucrados:** Usuario, Comprador.



- **Flujo Principal:**
  - El usuario accede al formulario de inicio de sesión.
  - El usuario ingresa sus credenciales (correo electrónico y contraseña).
  - El sistema valida las credenciales.
  - El sistema solicita un código de verificación adicional (enviado por SMS o correo electrónico).
  - El usuario ingresa el código de verificación.
  - El sistema verifica el código 2FA.
  - El sistema autentica al usuario y lo redirige a su área de página principal.

### Recuperar Contraseña CU03

- **Descripción:** Permite al usuario recuperar su contraseña en caso de olvido.
- **Actores involucrados:** Usuario.
- **Flujo Principal:**
  - El usuario selecciona la opción de "Recuperar contraseña".
  - El usuario ingresa su correo electrónico registrado.
  - El sistema envía un enlace de recuperación de contraseña al correo electrónico.
  - El usuario accede al enlace y restablecer su contraseña.
  - El sistema valida y confirma el restablecimiento de la contraseña.
  - El usuario puede iniciar sesión con la nueva contraseña.

### Generar Reportes de Ventas CU04

- **Descripción:** Permite al administrador generar reportes de ventas.
- **Actores involucrados:** Administrador.
- **Flujo Principal:**
  - El administrador accede al dashboard de administración.
  - El administrador selecciona la opción para generar reportes.
  - El sistema presenta opciones de filtros para los reportes (fecha, categoría de productos, etc.).
  - El administrador genera el reporte.
  - El sistema exporta el reporte en el formato solicitado





## Despliegue y Mantenimiento

### Despliegue

Para el despliegue de la aplicación abra en visual studio code donde clonó el repositorio del proyecto y siga las siguientes instrucciones

#### Frontend

instalar las dependencias con npm i  
npm run start

The screenshot shows the Visual Studio Code interface with the 'productos.controller.js' file open. The file contains JavaScript code for handling product requests. The Explorer sidebar on the left shows the project structure, including 'src', 'controllers', 'routes', and 'models'. The Terminal at the bottom shows the command 'npm run dev' being executed, which starts the development server on port 3200. The output of the command is visible in the terminal, showing the server starting and the database connection being established.

```
const filtrarProductos = async (req, res) => {  
  const productosConDetalles = await Promise.all(  
    productos.map(async (producto) => {  
      const detalles = await Producto.findById(producto.id);  
      return { producto, detalles };  
    })  
  );  
  return res.json({ ok: true, productos: productosConDetalles });  
} catch (error) {  
  await manejarErrores(error, res, "Producto");  
}
```

```
npm - Backend + ...  
devRandy  
D:\Documentos\Universidad\Octavo\AvD\E-commerce\Backend > 0.0 devRandy npm run dev  
backends1.0.0 dev  
nodemon src/index.js  
(nodemon) 3.1.4  
(nodemon) to restart at any time, enter 'rs'  
(nodemon) watching path(s): *.*  
(nodemon) watching extensions: js,mjs,cjs,json  
(nodemon) starting 'node src/index.js'  
Servidor iniciado en el puerto 3200  
[[12- Conexión establecida exitosamente con la base de datos: ecommerce  
Tablas sincronizadas con la base de datos.
```

para verificar que está funcionando correctamente puede realizar solicitudes desde postman

#### Backend

Instrucción mvn clean install