

Taller 3 - Apartamento 2D

Javier Leonardo Pinto
Fundación Universitaria Konrad Lorenz.
Diseño de Interfaces

Abstract—El proyecto de Diseño de interfaces se completa con la entrega de un prototipo de un apartamento funcional inteligente con detección de voz y detección de perros y humanos. El proyecto necesitó de herramientas digitales como HTML, CSS, JavaScript e incluso Photoshop.

I. INTRODUCCIÓN

En grandes partes del país somos concientes de que no existe una manera intuitiva de poder ver nuestro apartamento inteligente, aun más es muy notable que estas no cuentan con una detección de voz, esto es importante debido a que este permite ser más accesible a la hora de poder manejar nuestro apartamento inteligente. Además, es importante tener en cuenta que en muchos apartamentos se tienen perros así que detectarlos es muy importante.

Por lo que me abarqué en el proceso de aprender como crear esta interfaz que permitiera esa accesibilidad y este fue el resultado, una interfaz completa que permití controlar la casa inteligente.

II. METODOLOGÍA

Para realizar este apartamento utilicé diferentes herramientas digitales. Entre ellas, principalmente se usó HTML, CSS y JavaScript para poder crear el apartamento, sus espacios, sus formas visuales y sus funciones. También utilicé GitHub para subir los cambios, tener un historial del proceso y sus versiones y además tener un servicio de hosteo para tener mi proyecto en línea de forma gratuita y se utilizó Photoshop para crear las imágenes del apartamento.

III. IMPLEMENTACIÓN

Se implementó usando div, estilos en css, selectores ID, y constantes en el código de JavaScript.

Se utilizó ChatGPT para corrección de errores en el código

A. Visualización del plano del apartamento

Para lograr la visualización del apartamento se usaron muchos div en el código html y se hicieron clickeables de esta forma se podían ver las animaciones cuando las puestas, ventanas y luces estaban activadas o desactivadas.

Para las luces, ventanas y puertas se utilizaron div que cambiaban con respecto sus estilos.

B. Control con comando de voz

Para el comando de voz se utilizó lo visto en clase usando webkitSpeechRecognition ya integrado en el navegador.

Se le agrego un console.log para saber en consola lo que se dice.

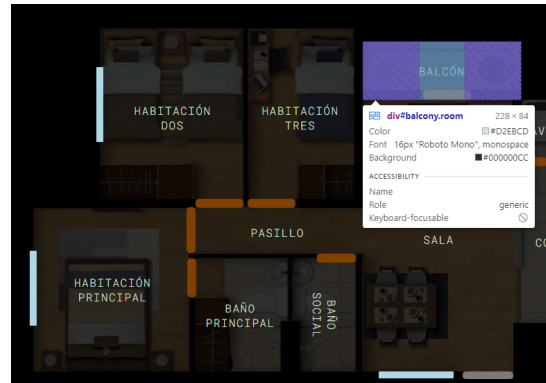


Fig. 1. Imagen del div

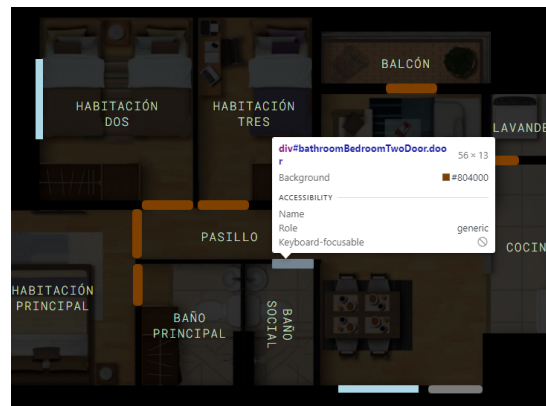


Fig. 2. Imagen de una puerta

C. Detección de perros o personas en la puerta principal

Para la detección de perros utilicé una función que detecta cuando aparece un perro o una persona e incluso ambas y en esta se genera un popup que le dice al usuario que detectó.

IV. RESULTADOS

En los resultados tenemos un apartamento que funciona perfectamente con luces, puertas y ventanas con animación y una cámara con detección de perro y humano.

V. CONCLUSIONES

Se cumplió con todos los requisitos el proyecto y además de ser intuitivo y accesible es funcional.

El proyecto permitió al usuario ver y controlar su apartamento inteligente.

<code>SpeechRecognition {grammars: SpeechGrammarList, lang:</code>	index.js:162
<code>▶ '', continuous: false, interimResults: false, maxAlternatives: 1, ...}</code>	
<code>SpeechRecognitionEvent {isTrusted: true, resultIndex: 0,</code>	index.js:169
<code>▶ results: SpeechRecognitionResultList, type: 'result', target: SpeechRecognition, ...}</code>	
<code>prende la luz de la sala</code>	index.js:181
<code>add</code>	index.js:269
<code>light</code>	index.js:270
<code>livingRoom</code>	index.js:271
<code>Recognized text: prende la luz de la sala</code>	index.js:281
<code>></code>	

Fig. 3. Imagen del reconocimiento de voz



Fig. 4. Imagen de la detección de perro y humano.



Fig. 5. Conclusión

VI. BIBLIOGRAFÍA

REFERENCES

[1] OpenAI. "ChatGPT." OpenAI, 2021. En línea. Disponible en: <https://openai.com/>. Consultado el 3 de junio de 2023.