

Jornadas de Automática

Plataforma web modular para terapias de pacientes de atención temprana con autismo asistidas por el robot NAO

Gómez-Espínosa, A.^{a,1,*}, Moreno-Úbeda, J.C.^b, Blanco-Claraco, J.L.^a

^aUniversity of Almería, Department of Engineering, CIESOL, ceia3, Ctra. Sacramento s/n, 04120, Almería, Spain, {agespinosa, jlblanco}@ual.es

^bUniversity of Almería, Department of Informatics, CIESOL, ceia3, Ctra. Sacramento s/n, 04120, Almería, Spain, {jcmoreno, juandiego.gil}@ual.es

To cite this article: Gómez-Espínosa, A., Moreno-Úbeda, J. C., Blanco-Claraco, J. L., 2026. Modular web platform for early intervention therapies for patients with autism, assisted by the NAO robot. *Jornadas de Automática*, 47. <https://doi.org/10.17979/ja-cea.2026.47.13870>

Resumen

Este artículo presenta el diseño y desarrollo de una aplicación web asistencial orientada a optimizar el uso del robot humanoide NAO en terapias de atención temprana para niños con Trastorno del Espectro Autista (TEA). Se propone una arquitectura de *monorepo* desacoplada basada en *Next.js*, *Node.js* y *Flask*. El sistema destaca por integrar un motor adaptativo que ajusta la dificultad de la terapia de forma dinámica (entre 3 y 9 estímulos) según el histórico del rendimiento del paciente. Asimismo, incorpora un mecanismo de bloqueo de la interfaz durante la locución del robot para promover la escucha activa y evitar la impulsividad. La plataforma permite el registro longitudinal de datos de interacción y, mediante un despliegue simplificado con *Docker*, facilita su adopción por personal clínico sin perfil técnico, reduciendo la brecha entre el potencial de la Robótica Social Asistencial y su aplicación en la práctica clínica real.

Palabras clave: Robótica de asistencia social, Interacción humano-robot, Control adaptativo de la dificultad, Interfaz de teleoperación basada en web, TEA

Modular web platform for early intervention therapies for patients with autism, assisted by the NAO robot.

Abstract

This article presents the design and development of a support web application aimed at optimising the use of the NAO humanoid robot in early intervention therapies for children with Autism Spectrum Disorder (ASD). To overcome the limitations of traditional programming, a decoupled monorepo architecture is proposed, based on Next.js, Node.js and Flask. The system stands out for integrating an adaptive engine that dynamically adjusts the difficulty level for the therapy (between 3 and 9 stimuli) based on the patient's real time performance. It also incorporates a mechanism to lock the interface whilst the robot is speaking to promote active listening and prevent impulsivity. The platform enables the longitudinal recording of interaction data and, through simplified deployment with Docker, facilitates its adoption by clinical staff without a technical background, bridging the gap between the potential of Socially Assistive Robotics and its application in real-world clinical practice.

Keywords: Socially Assistive Robotics (SAR), Human-Robot Interaction, Adaptive difficulty control, Web-based teleoperation interface, NAO

*Autor para correspondencia: agespinosa@ual.es

Attribution-NonCommercial-ShareAlike 4.0 International (CC BY-NC-SA 4.0)

Correos electrónicos: agespinosa@ual.es (Gómez-Espínosa, A.), jcmoreno@ual.es (Moreno-Úbeda, J.C.), jlblanco@ual.es (Blanco-Claraco, J.L.)

URL: www.cea-ifac.es (Gómez-Espínosa, A.)

1. Introducción

El Trastorno del Espectro Autista (TEA) es una condición del neurodesarrollo caracterizada por déficits persistentes en la comunicación social y por patrones de conducta, intereses y actividades restringidos o repetitivos (Alshahrani, 2025; Di Cara et al., 2025). Su prevalencia ha aumentado de forma sostenida en las últimas décadas, ya que las estimaciones recientes sitúan la incidencia en torno a 1 de cada 54 niños a nivel mundial (Alshahrani, 2025) y en torno al 1,5 % de la población infantil en España (Lorenzo Lledó et al., 2024); además, entre el 25 % y el 30 % presentan un perfil mínimamente verbal (Di Cara et al., 2025). Las intervenciones tradicionales de atención temprana, como Análisis Conductual Aplicado (ABA), logopedia, Sistemas de Comunicación Aumentativa y Alternativa (CAA) o entrenamiento en habilidades sociales, han mostrado eficacia, pero su éxito depende de una alta personalización, de profesionales especializados y de un coste elevado (Di Cara et al., 2025; Alnafjan et al., 2025), lo que ha impulsado la búsqueda de herramientas tecnológicas que complementen el trabajo clínico, aumenten la adherencia y aporten entornos predecibles (Lorenzo Lledó et al., 2024; Biagi et al., 2025; Gómez-Espinosa et al., 2024).

En este escenario, la robótica asistencial social (SAR, del inglés *Socially Assistive Robotics*) se ha consolidado como una de las líneas más prometedoras dentro de las terapias con niños TEA (Alnafjan et al., 2025; Bettencourt et al., 2025; Gómez-Espinosa et al., 2024). Los niños con TEA tienden a orientarse antes hacia objetos no sociales que hacia personas, ya que perciben a los robots como entes más estereotipados y predecibles (Biagi et al., 2025). Esta predictibilidad reduce la ansiedad y favorece la aparición espontánea de contacto visual, atención conjunta, imitación o iniciativa comunicativa (Di Cara et al., 2025; Alnafjan et al., 2025; Zorcec et al., 2021). Entre los robots empleados, Kaspar, ZENO, Probo, Cozmo, QT o Pepper, son algunos ejemplos, el humanoide NAO es el más usado en las intervenciones con resultados publicados en la última década (Mutawa et al., 2023; Biagi et al., 2025; Gómez-Espinosa et al., 2024; Di Cara et al., 2025).

Sin embargo, la adopción de NAO en la práctica clínica tropieza con la barrera transversal de la complejidad de su programación. El entorno oficial *Choregraphe* exige conocimientos técnicos avanzados y sus flujos visuales resultan inmanejables al crecer la rutina (Biagi et al., 2025; Leaf et al., 2017), por lo que el terapeuta depende a menudo de un técnico, lo que ralentiza la adaptación de las actividades y limita la escalabilidad de las intervenciones (Elbeleidy et al., 2021; Leaf et al., 2017). De hecho, la mayoría de las terapias asistidas por robots en clínica real operan en esquemas de teleoperación tipo *Wizard-of-Oz*, en los que el terapeuta selecciona en tiempo real qué dice o hace el robot (Elbeleidy et al., 2021; Bettencourt et al., 2025). Mejorar la usabilidad de las interfaces de control se ha identificado de forma reiterada como un requisito crítico para el despliegue efectivo del NAO en consulta (Biagi et al., 2025; Elbeleidy et al., 2021).

En paralelo, el ecosistema de aplicaciones móviles y web orientadas al TEA está en plena expansión (Zorcec et al., 2021; Abu-Amara et al., 2024), y su combinación con un robot social ha mostrado resultados positivos. En este caso la app actúa como interfaz pedagógica y de control mientras el

robot opera como mediador social (Zorcec et al., 2021; Mutawa et al., 2023; Abu-Amara et al., 2024; Azhar et al., 2021). En Mutawa et al. (2023) los autores integraron NAO con una app móvil y observaron mayor contacto visual e implicación de los niños, por otro lado, en Abu-Amara et al. (2024) entrenaron habilidades académicas con NAO y la aplicación *AutisPlay*. Además, en Zorcec et al. (2021) se evidenciaron mejoras en imitación, juego y atención al combinar Kaspar con una app complementaria y en Azhar et al. (2021) construyeron una suite educativa que orquesta NAO y Cozmo desde una app iOS. En el plano técnico, distintos autores han propuesto arquitecturas basadas en *WebSockets* o *HTTP* para que tabletas, smartphones o navegadores envíen comandos al robot en tiempo real, evitando los inconvenientes de *Choregraphe* (Leaf et al., 2017; Azhar et al., 2021). Más recientemente, en Biagi et al. (2025) se han validado dos modos de control alternativos (*Puppet* y *Assistant*, este último basado en Modelos de Lenguaje de Gran Escala), confirmando que los clínicos prefieren claramente interfaces sencillas frente a la programación tradicional del robot.

Pese a estos avances, persisten varias carencias. En primer lugar, la mayoría de soluciones se diseña para un único robot o un contexto muy específico, lo que dificulta su generalización a las seis grandes áreas habituales de la atención temprana en TEA (Abu-Amara et al., 2024; Zorcec et al., 2021): colores, animales, formas, medios de transporte, números y partes del cuerpo. En segundo lugar, son escasos los sistemas con ajuste multinivel automático de la dificultad basado en el rendimiento histórico, factor decisivo en la eficacia terapéutica (Di Cara et al., 2025; Alshahrani, 2025). En tercer lugar, pocos abordan el *bloqueo de la interfaz* mientras el robot habla, condición indispensable para favorecer la escucha activa en niños con elevada impulsividad. Por último, casi ninguna propuesta integra simultáneamente la gestión multiperfil del terapeuta, el registro longitudinal de cada iteración (aciertos, fallos y tiempo de respuesta) y la visualización de estadísticas que permitan al profesional decidir basándose en datos (Azhar et al., 2021; Mutawa et al., 2023).

A partir de estas evidencias, el presente trabajo propone el desarrollo de una aplicación web de apoyo a las terapias de atención temprana en niños con TEA, que actúe como capa de control intuitiva sobre el robot NAO. La aplicación está concebida para que logopedas, psicólogos y terapeutas ocupacionales integren al robot en su práctica sin necesidad de conocimientos de programación, delegando en el robot NAO la enunciación verbal de las consignas mientras la web gestiona los estímulos visuales, el registro de la interacción y el ajuste dinámico de la dificultad. La arquitectura se apoya en un *stack* tecnológico moderno (*React/Next.js*, *Node.js/Express* y base de datos relacional y documental) y en una comunicación bidireccional con el robot mediante *API REST* o *WebSockets*, conforme a las recomendaciones técnicas más recientes (Leaf et al., 2017; Azhar et al., 2021; Biagi et al., 2025). El sistema contempla seis categorías terapéuticas, cuatro niveles de dificultad basados en el número de distractores en pantalla (3, 5, 7 y 9 estímulos) y un mecanismo automático de progresión multinivel calculado a partir del porcentaje de aciertos de las tres últimas sesiones del niño en cada categoría. La interfaz incorpora principios estrictos de accesibilidad cognitiva y diseño inclusivo, evita la sobreestimulación sensorial y bloquea los

botones de respuesta mientras el robot está hablando, protegiendo el objetivo terapéutico de la escucha activa. El objetivo general es, por tanto, facilitar a los terapeutas el uso del robot NAO mediante una herramienta web que combine usabilidad, personalización dinámica de la dificultad, registro objetivo del progreso e integración sencilla con el robot, contribuyendo a reducir la brecha entre el potencial demostrado por la SAR en el TEA y su adopción efectiva en la práctica clínica diaria.

2. Visión general de la arquitectura de la aplicación

2.1. Estilo arquitectónico y principios rectores

El sistema se compone en sí por cuatro entidades o niveles lógicos de responsabilidad llamadas capas. En este caso para la generación de la aplicación se han creado las capas de presentación, los dominios y la presencia, y se ha añadido el robot NAO, que es nuestra cuarta entidad, como agente físico.

1. **Capa de presentación:** Una aplicación web que asume la doble función de panel clínico para el terapeuta y de superficie de interacción para el niño. Esta capa se limita a presentar estado y a emitir intenciones.
2. **Capa de dominio:** Un servicio de aplicación que concentra las reglas terapéuticas, el motor de adaptación y la autenticación.
3. **Capa de persistencia:** Un sistema de gestión de bases de datos relacional que sostiene la integridad transaccional sobre el dominio (terapeutas, niños, sesiones, interacciones).
4. **Agente físico.** El robot NAO, expuesto mediante un servicio de adaptación.

La capa de dominio y la capa de persistencia pertenecen al servidor de la aplicación, es decir, son dos capas independientes que forman parte del *Backend* de nuestra aplicación. El *backend* se encarga de todo lo que ocurre en el lado del servidor, incluyendo el procesamiento de solicitudes, la gestión de datos y la seguridad de la aplicación.

La arquitectura desarrollada se realizó mediante estilo de *arquitectura por capas* con responsabilidades estrictamente separadas y aplica el principio de *Clean Architecture*, es decir, las capas exteriores (presentación, persistencia y el agente físico) dependen de las interiores (dominio) y nunca al revés. Cada capa se materializa como un paquete del *monorepo* (repositorio único), lo que permite versionarla, probarla y desplegarla con independencia del resto. Estas capas se representan en la Figura 1.

Los usuarios (terapeuta y niño) interactúan con una única interfaz, la interfaz del cliente web. El cliente es el programa que consume un servicio remoto. En nuestro caso, es la aplicación web que se ejecuta dentro del navegador (Chrome, Edge, Firefox...). Cuando esa aplicación necesita datos, por ejemplo, la lista de niños o el nivel actual de una sesión, se los pide al servidor, que es el programa que ofrece el servicio. En este caso vamos a tener dos tipos de usuarios distintos, uno es el terapeuta que usa el panel clínico, el otro será el niño que usa la vista de juego.

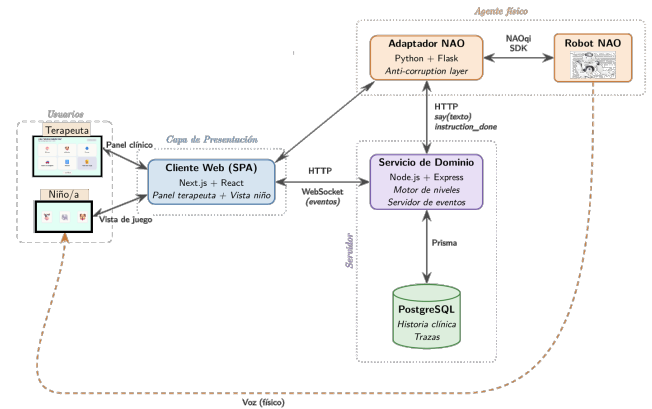


Figura 1: Arquitectura de la aplicación

El cliente no contiene reglas terapéuticas, si no que todo lo que necesita decidir se lo pide al servicio de dominio. Se comunica con él por dos canales distintos. Usa HTTP para operaciones puntuales y WebSocket para eventos en tiempo real. Estas conexiones son bilaterales, es decir, el cliente Web pide datos y el servidor de dominio se los envía.

El servicio de dominio es el núcleo del sistema. Aplica el motor de niveles, valida sesiones, persiste datos en PostgreSQL y da ordenes al robot a través del adaptador. El adaptador NAO aísla por completo el SDK propietario del robot. El SDK al residir en el robot, se arranca por separado siguiendo el patrón ya descrito por Leaf et al. (2017). Si mañana se sustituye el NAO por otro agente social, sólo cambia este componente. Por último, el robot NAO entrega la instrucción al niño por una vía no digital (usa la voz) que en el esquema aparece como flecha discontinua. Es la única conexión *física* del sistema.

3. Capa de presentación

La capa de presentación contiene un único componente (Cliente Web (SPA)) construido con Next.js y React. Es la cara visible del sistema. Sirve dos roles distintos sobre la misma base de código: el panel clínico para el terapeuta (Figura 2) y la vista de juego para el niño (Figura 3). No toma decisiones terapéuticas, solo traduce intenciones del usuario en peticiones al servidor y refleja en pantalla el estado que el servidor le comunica. La selección de las seis categorías terapéuticas (colores, animales, formas geométricas, medios de transporte, números y partes del cuerpo) responde a los dominios de vocabulario receptivo y conceptual que los programas de atención temprana para niños con TEA trabajan de forma prioritaria en las primeras etapas de intervención (Lorenzo Lledó et al., 2024). Estas categorías aparecen de forma recurrente en propuestas comparables de robótica asistencial: Abu-Amara et al. (Abu-Amara et al., 2024) las emplean para el entrenamiento de habilidades académicas con NAO, y Zorcec et al. (Zorcec et al., 2021) incluyen categorías equivalentes en sus sesiones con Kaspar. Su carácter concreto y visualmente discriminable las hace especialmente adecuadas para la modalidad de señalización por imagen utilizada en la interfaz, minimizando la carga cognitiva y favoreciendo la generalización de los aprendizajes (Alnafjan et al., 2025).

3.1. Accesibilidad cognitiva

Siguiendo los principios de diseño inclusivo para neurodiversidad Gilbert (2019), la pantalla del niño aplica una paleta pastel sin blancos puros, tipografía *Atkinson Hyperlegible*, bordes redondeados, cuadrícula simétrica para los estímulos y ausencia de castigo visual ante un fallo (la opción incorrecta se desvanece en vez de marcarse en rojo). El feedback positivo se limita a una animación de borde verde suave, sin parpadeos ni sonidos estridentes. Estas decisiones, alineadas con la literatura de SAR para TEA (Di Cara et al., 2025; Alnafjan et al., 2025), se implementan en componentes específicos (*StimulusCard*, *FeedbackAnimation*) y son parametrizables desde un único módulo de constantes (`lib/constants.ts`).

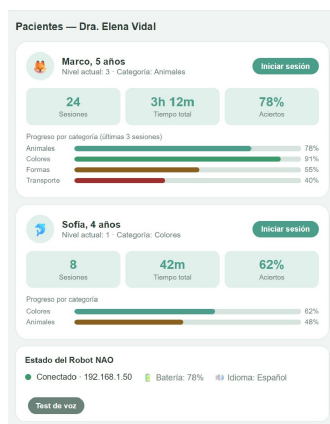


Figura 2: Pantalla de datos de seguimiento de la App

3.2. Bloqueo de interfaz durante la instrucción verbal

El componente *BlockingOverlay* cubre la cuadrícula de estímulos mientras NAO está hablando. Se activa al enviarse la petición de instrucción y se retira tras después de que el robot haya terminado. Este mecanismo de bloqueo es deliberadamente *positivo* (no es una excepción de error) y constituye una salvaguarda clínica frente a la impulsividad típica del niño con TEA, alineada con la recomendación reiterada en la literatura de proteger el objetivo de la escucha activa (Elbeleidy et al., 2021; Biagi et al., 2025).

4. Capa de persistencia

La capa de persistencia constituye el sustrato de memoria a largo plazo de la aplicación. Es la responsable de conservar la información clínica del paciente, el registro de los profesionales que lo atienden y la traza exhaustiva de cada sesión terapéutica realizada. Se sustenta en un *sistema de gestión de bases de datos relacional* denominado *PostgreSQL*. PostgreSQL es un proyecto consolidado, de código abierto, con más de tres décadas de desarrollo continuado y una reputación bien establecida en entornos donde la fiabilidad de los datos es crítica (incluyendo aplicaciones científicas, financieras y sanitarias).

Cada dato que el dominio confía a esta capa queda registrado de manera duradera y verificable. Si el sistema se interrumpe inesperadamente (por un corte eléctrico, por un cierre forzoso del programa, por un fallo de red), la información que

ya estaba registrada no se pierde y la que estaba a medio escribir no queda en un estado ambiguo: o se completó correctamente, o se descarta por completo. Esta garantía, conocida como *integridad transaccional*, es esencial para que el historial clínico merezca confianza.

La información se organiza en *tablas* interconectadas que representan las entidades del dominio terapéutico, lo que quiere decir, una tabla de terapeutas, una de niños, una de sesiones, una de interacciones y una pequeña tabla de progreso. Cada niño está formalmente vinculado a su terapeuta; cada sesión, a su niño y a su categoría; cada interacción, a su sesión. Estas conexiones explícitas impiden que aparezcan, por error, registros huérfanos o contradictorios.

La capa permite consultar la información de forma muy precisa. Las tres últimas sesiones de un niño en la categoría de animales, el tiempo total empleado por un paciente, la tasa de aciertos en un periodo, la lista de niños asignados a un terapeuta. Estas consultas se resuelven en tiempo prácticamente instantáneo gracias a estructuras internas denominadas *índices*, que actúan como las pestañas de un archivador físico. Además, no sobrescribe información clínica, es decir, cada interacción del niño con el sistema queda registrada como un evento distinto, con su fecha y su contexto. El historial es, por tanto, un *registro acumulativo y auditable*, sobre el cual pueden realizarse análisis longitudinales fiables.

4.1. Justificación frente a alternativas documentales

La elección de un motor relacional, frente a alternativas documentales (*MongoDB*, *Firestore*), es debido al modo intrínsecamente relacional de los datos clínicos y a la necesidad de garantizar integridad referencial entre terapeuta, niño, sesión e interacción. Patrones similares se observan en sistemas comparables (Azhar et al., 2021; Mutawa et al., 2023), con un coste operativo bajo gracias al despliegue del motor como contenedor *Docker*.

5. Capa de dominios

La capa de dominio se ha implementado sobre un entorno de ejecución ampliamente utilizado en el desarrollo de aplicaciones web modernas, denominado *Node.js*, complementado con un marco de trabajo llamado *Express* que organiza la recepción y respuesta de peticiones de red. Para comunicarse con la base de datos, el componente que persiste la información de forma duradera, el dominio emplea *Prisma*, que actúa como traductor entre el modelo de información del dominio y el almacenamiento relacional.

Para los eventos en tiempo real, los que sostienen el bloqueo y desbloqueo de la pantalla durante la verbalización del robot, el dominio incorpora un componente denominado *Socket.io*, que mantiene abierto un canal de comunicación permanente con la interfaz. Sin este componente, sería imposible garantizar la sincronía entre lo que el robot dice y lo que la pantalla permite.

5.1. Motor de niveles adaptativo

La personalización dinámica es uno de los factores con mayor impacto en la eficacia de la terapia (Di Cara et al., 2025; Alshahrani, 2025), pero su implementación rara vez se

documenta de manera transparente. En la aplicación el nivel determina el número de estímulos visibles en la cuadrícula, $L_1=3$, $L_2=5$, $L_3=7$ y $L_4=9$, y se recalcula al cerrar cada sesión a partir de las tres últimas sesiones del niño en la categoría correspondiente.

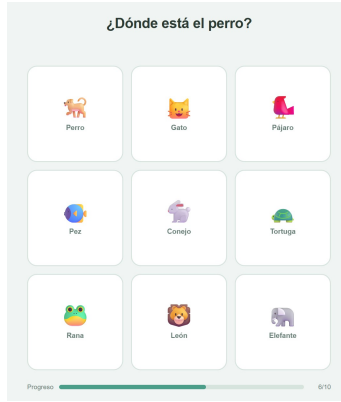


Figura 3: Cuadrícula del nivel 4 ($L_4=9$ estímulos): los nueve distractores se distribuyen simétricamente para evitar la sobreestimulación al tiempo que se mantiene la dificultad terapéutica.

Sea a la tasa media de aciertos en esa ventana, calculada como

$$a = \frac{\sum_{i=1}^N \text{aciertos}_i}{\sum_{i=1}^N (\text{aciertos}_i + \text{fallos}_i)}, \quad N = \text{mín}(3, \text{siones acabadas}),$$

el siguiente nivel se obtiene como

$$L_{n+1} = \begin{cases} \text{mín}(L_n + 1, 4) & \text{si } a \geq 0,80, \\ \text{máx}(L_n - 1, 1) & \text{si } a \leq 0,40, \\ L_n & \text{en otro caso.} \end{cases}$$

Si el niño no dispone de sesiones cerradas en esa categoría se inicializa en L_1 .

La función anterior es pura, no escribe en base de datos, lo que la hace verificable mediante pruebas unitarias. Esta se invoca al cerrar una sesión y se consulta al abrir la siguiente. Esta lógica explícita y auditable contrasta con la opacidad habitual de las reglas de progresión en otras propuestas integradas (Abu-Amara et al., 2024; Zorcec et al., 2021).

Los umbrales de progresión elegidos ($a \geq 0,80$ para subir de nivel y $a \leq 0,40$ para bajar) tienen una doble justificación. Desde la perspectiva de la intervención conductual, el criterio de dominio del 80 % de respuestas correctas es un estándar ampliamente consolidado en los programas de Análisis Conductual Aplicado (ABA) para considerar adquirida una habilidad antes de introducir mayor complejidad (Cooper et al., 2020). El umbral inferior del 40 % actúa como señal de retroceso cuando el rendimiento cae por debajo de lo que puede atribuirse al azar en una cuadrícula de tres estímulos ($p = 0,33$), indicando que el nivel actual supera las capacidades actuales del niño. Desde la perspectiva del diseño de experiencias de aprendizaje, ambos umbrales delimitan la banda de dificultad óptima descrita por la teoría del flujo (Csikszentmihalyi, 1990): cuando la dificultad de la tarea está equilibrada con la habilidad del usuario se favorece el compromiso sostenido, evitando tanto la ansiedad asociada a tareas excesivamente difíciles como el aburrimiento derivado de tareas

demasiado sencillas, lo que resulta especialmente relevante en poblaciones con TEA donde la regulación emocional ante el fracaso es un factor crítico (Di Cara et al., 2025).

6. Integración de la aplicación con el robot

La comunicación entre el servidor y el robot adopta una arquitectura híbrida. Cuando el terapeuta inicia una interacción, el servicio de dominio construye en castellano el texto de la consigna a partir de plantillas por categoría definidas, por ejemplo, “toca el color rojo” o “¿dónde está el perro?”, y lo envía mediante HTTP al adaptador del robot. El adaptador llama a `ALTextToSpeech.say`, llamada *bloqueante* del NAOqi, es decir, la respuesta HTTP sólo regresa cuando NAO ha terminado de hablar, evitando por construcción el uso de *timers* o sondeo y proporcionando al servicio de dominios el instante exacto en que la consigna ha concluido.

Inmediatamente después, el servidor emite por *Socket* el evento, que el cliente escucha para retirar el componente bloqueante y habilitar las respuestas al niño. Este *bloqueo de la interfaz mientras el robot habla* es una salvaguarda clínica explícita, es decir, en niños con elevada impulsividad protege el objetivo terapéutico de la escucha activa, recomendación rara vez incorporada en los sistemas previos revisados (Elbeleidy et al., 2021; Biagi et al., 2025). Adicionalmente, un latido periódico (cada 10 s) consulta y propaga el estado de conexión del robot, siguiendo las buenas prácticas de teleoperación documentadas por Leaf et al. (2017).

6.1. Empaquetado y despliegue para personal no técnico

El *monorepo* se gestiona con *pnpm* 8 y se versiona en un único repositorio. La capa de datos se aprovisiona con *docker compose up -d postgres*, eliminando la necesidad de instalar *PostgreSQL* en el equipo del terapeuta. Para el usuario final, profesional sin perfil técnico, se distribuye un acceso directo de Windows que invoca un *script* de arranque. Dicho *script* comprueba secuencialmente la disponibilidad de *Docker*, arranca el contenedor de base de datos, aplica las migraciones de Prisma de forma idempotente, programa la apertura automática del navegador en `http://localhost:3000` y lanza el *monorepo* en modo desarrollo. Cerrar la única ventana de consola detiene todos los servicios. Esta capa de despliegue, frecuentemente ignorada en los prototipos académicos, responde directamente a la barrera de adopción identificada en la literatura (Biagi et al., 2025; Elbeleidy et al., 2021).

7. Conclusión, discusión y trabajo futuro

En este artículo se ha presentado el diseño e implementación de una aplicación web de apoyo a las terapias de atención temprana en niños con TEA, concebida como interfaz de control intuitiva sobre el robot humanoide NAO. El objetivo central ha sido reducir la brecha existente entre el potencial demostrado por la SAR y su adopción efectiva en la práctica clínica, articulando cuatro contribuciones principales.

La organización del sistema como *monorepo* con tres capas separadas (presentación, dominio y persistencia), permite que el terapeuta opere el robot sin conocimiento de programación. Delegar la enunciación verbal al robot y la gestión de estímulos visuales a la interfaz web responde directamente a

la barrera de usabilidad de Choregraphe (Biagi et al., 2025; Elbeleidy et al., 2021). El patrón *Backend for Frontend*, junto con la comunicación híbrida *REST/WebSocket*, garantiza un único punto de control y auditoría, facilitando la trazabilidad clínica de cada sesión.

El sistema implementa un mecanismo de progresión multinivel, cuatro niveles de dificultad determinados por el número de distractores en pantalla, cuya regla de transición es explícita y auditable. El nivel se recalcula al cerrar cada sesión a partir de la tasa media de aciertos en las tres últimas sesiones del niño en cada una de las seis áreas terapéuticas cubiertas. Esta transparencia contrasta con la opacidad frecuente en propuestas comparables (Abu-Amara et al., 2024; Zorcec et al., 2021) y refuerza la confianza del profesional en la herramienta.

El mecanismo de bloqueo de interfaz durante la enunciación verbal del robot constituye una aportación de diseño con justificación terapéutica explícita. Al inhabilitar los botones de respuesta mientras NAO habla, habilitándolos únicamente tras recibir el evento *WebSocket*, el sistema protege el objetivo de la escucha activa en niños con elevada impulsividad (Elbeleidy et al., 2021; Biagi et al., 2025). Combinado con un diseño de pantalla conforme a principios de accesibilidad cognitiva (paleta pastel, tipografía *Atkinson Hyperlegible*, ausencia de castigo visual y cuadrícula simétrica), se minimiza la sobreestimulación sensorial (Di Cara et al., 2025; Alnafjan et al., 2025).

El modelo relacional (PostgreSQL + Prisma) captura cada iteración con el estímulo objetivo, el elegido, el resultado, el tiempo de respuesta y el instante en que NAO concluyó la consigna. Los índices diseñados permiten generar en tiempo real gráficos longitudinales que el terapeuta consulta desde el *dashboard* para tomar decisiones basadas en datos, cubriendo una carencia identificada en la mayoría de las propuestas previas (Azhar et al., 2021; Mutawa et al., 2023).

El presente trabajo constituye una propuesta de sistema técnicamente fundamentada, cuya validación clínica con terapeutas y pacientes reales queda como trabajo futuro prioritario. La eficacia clínica requiere estudios controlados con participantes reales y métricas validadas (contacto visual, atención conjunta, generalización). El adaptador está vinculado al *NAOqi*, limitando la portabilidad a otras plataformas robóticas; su sustitución por una capa de abstracción genérica es una línea de trabajo futuro. Aunque la arquitectura por capas aísla la dependencia del SDK propietario en el adaptador NAO, la portabilidad a otros robots está condicionada a que la plataforma destino ofrezca una API de síntesis de voz bloqueante equivalente y acceso programático a sus actuadores. La sustitución del adaptador es, por tanto, posible en principio, pero no trivial, y constituye una línea de trabajo futura que requerirá validación específica para cada plataforma. Además, como uno de los trabajos futuros más cercano, sería aumentar la variabilidad de terapias, para poder adaptar las sesiones a necesidades más específicas del paciente. Como extensiones funcionales se contempla incorporar análisis de expresión facial mediante visión por computador para enriquecer el perfil de progreso del niño, así como un módulo de comunicación familiar que permita reforzar en el hogar las habilidades trabajadas en sesión.

En conclusión, la aplicación propuesta demuestra que es posible construir una herramienta de SAR clínicamente mo-

tivada, técnicamente sólida y accesible para profesionales sin perfil técnico, contribuyendo a hacer efectivo el potencial terapéutico del robot NAO en la intervención temprana con niños con TEA.

Referencias

- Abu-Amara, F., Mohammad, H., Bensefia, A., 2024. Robot-based therapy for improving academic skills of children with autism. *International Journal of Information Technology* 16 (6), 3371–3380.
DOI: 10.1007/s41870-024-01883-1
- Alnafjan, A., Alghamdi, M., Alhakhani, N., Al-Ohali, Y., 2025. Improving imitation skills in children with autism spectrum disorder using the NAO robot and a human action recognition. *Diagnostics* 15 (1), 60.
DOI: 10.3390/diagnostics15010060
- Alshahrani, S. M., 2025. The impact of AI-mediated therapy on treatment outcomes in autism spectrum disorder: A systematic review. *Journal of Evidence-Based Psychotherapies* 25 (2), 71–102.
DOI: 10.24193/jebp.2025.2.12
- Azhar, M. Q., Jiang, W., Schattner-Elmaleh, N., 2021. Collaboration education suite for children with ASD. Companion of the 2021 ACM/IEEE International Conference on Human-Robot Interaction (HRI '21 Companion), 439–441.
DOI: 10.1145/3434074.3447209
- Bettencourt, C., Grossard, C., Zou, J., Segretain, M., Bree, M., Pellerin, H., Anzalone, S. M., Chetouani, M., Cohen, D., 2025. Investigating the feasibility of a wizard-of-oz robotic interface (R2C3) in a social skills group for children with autism spectrum disorder. *International Journal of Social Robotics* 17, 1395–1411.
DOI: 10.1007/s12369-025-01243-4
- Biagi, F., Iani, C., Biagiotti, L., 2025. The use of the social robot NAO in medical settings: How to facilitate interactions between healthcare professionals and patients with autism spectrum disorder. *Frontiers in Psychiatry* 16, 1675098.
DOI: 10.3389/fpsy.2025.1675098
- Cooper, J. O., Heron, T. E., Heward, W. L., 2020. *Applied Behavior Analysis*, 3rd Edition. Pearson, Hoboken, NJ.
- Csikszentmihalyi, M., 1990. *Flow: The Psychology of Optimal Experience*. Harper & Row, New York.
- Di Cara, M., La Fauci, M., Tresoldi, M., Caputo, M. R., Borzelli, D., Maggio, R., Campestre, C., Barbera, A., Piccolo, A., De Domenico, C., Di Blasi, M., Calabrò, R. S., Tripodi, E., Impallomeni, C., Cucinotta, F., 2025. Enhancing communication in minimally verbal autistic children: A study on NAO-assisted therapy. *Journal of Clinical Medicine* 14 (11), 3735.
DOI: 10.3390/jcm14113735
- Elbeleidy, S., Rosen, D., Liu, D., Shick, A., Williams, T., 2021. Analyzing teleoperation interface usage of robots in therapy for children with autism. *Proceedings of the 20th ACM Interaction Design and Children Conference (IDC '21)*, 112–118.
DOI: 10.1145/3459990.3460715
- Gilbert, R. M., 2019. *Inclusive Design for a Digital*. Springer.
DOI: 10.1007/978-1-4842-5016-7
- Gómez-Espinosa, A., Moreno, J. C., Pérez-de la Cruz, S., 2024. Assisted robots in therapies for children with autism in early childhood. *Sensors* 24 (5), 1503.
DOI: 10.3390/s24051503
- Leaf, J., Gerlick, R. E., Richter, D. C., 2017. Remote interaction with a NAO robot using a tablet device. *Proceedings of the ASEE Annual Conference*, 10.
DOI: 10.18260/1-2--28798
- Lorenzo Lledó, G., Lorenzo Lledó, A., Rodríguez-Quevedo, Á., 2024. Análisis mediante inteligencia artificial de las emociones del alumnado autista en la interacción social con el robot NAO. *RED. Revista de Educación a Distancia* 24 (78).
DOI: 10.6018/red.588091
- Mutawa, A. M., Al Mudhahkah, H. M., Al-Huwais, A., Al-Khaldi, N., Al-Otaibi, R., Al-Ansari, A., 2023. Augmenting mobile app with NAO robot for autism education. *Machines* 11 (8), 833.
DOI: 10.3390/machines11080833
- Zorcec, T., Ilijoski, B., Simlesa, S., Ackovska, N., Rosandic, M., Popcevic, K., Robins, B., Nitzan, N., Cappel, D., Blum, R., 2021. Enriching human-robot interaction with mobile app in interventions of children with autism spectrum disorder. *Prilozi (Macedonian Academy of Sciences and Arts)* 42 (2).
DOI: 10.2478/prilozi-2021-0021