



# Interreg España - Portugal



Fondo Europeo de Desarrollo Regional  
Fundo Europeu de Desenvolvimento Regional



## Iniciativas innovadoras para el impulso del envejecimiento activo en la región EuroACE

### TÍTULO DOCUMENTO

**Selección de desarrollos y aplicaciones para la interacción humano-robot**

### RESPONSABLE:

**RoboLab**

### FECHA DE ENTREGA:

**31-09-2019**

### AUTORES:

**Pedro M. Núñez Trujillo**

### CONTRIBUCIONES:

**José Luis Moyano, Blas Pagador, Pedro Núñez, Paulo Gonçalves, Rui Rocha, Carolina Vila**

### RESUMEN DEL DOCUMENTO

Este documento describe el conjunto de desarrollos y aplicaciones desarrolladas en el marco del proyecto. Se describe la arquitectura cognitiva desarrollada CORTEX+ y los agentes para la interacción.

## Información del documento

FECHA	VERSIÓN	COMENTARIOS
15/09/2018	V0.0	Versión inicial
22/02/2019	V1.0	Versión con contribuciones de Pedro Núñez, Paulo Gonçalves, Rui Rocha
31/09/2019	V2.0	Versión final



Centro de Cirugía de Mínima Invasión  
Minimally Invasive Surgery Centre  
Jesús Usón

Fundación Centro de Cirugía de  
Mínima Invasión Jesús Usón  
(CCMIJU, España)

Coordinador:  
Jose Blas Pagador  
[jbpagador@ccmiiesusoson.com](mailto:jbpagador@ccmiiesusoson.com)



Instituto Politécnico  
de Castelo Branco

Instituto Politécnico de Portalegre.  
(IPCB, Portugal)

Contacto:  
Paulo Gonçalves  
[paulo.goncalves@ipcb.pt](mailto:paulo.goncalves@ipcb.pt)



Universidad de Coimbra (UC,  
Portugal)

Contacto:  
Rui Rocha  
[rprocha@deec.uc.pt](mailto:rprocha@deec.uc.pt)



Cluster de la Salud

Cluster Sociosanitario de  
Extremadura (Cluster, España)

Contacto:  
Nicolas Montero  
[gerente@clustersalud.es](mailto:gerente@clustersalud.es)



Universidad de Extremadura (UEX,  
España)

Contacto:  
Pedro Núñez  
[pnuntru@unex.es](mailto:pnuntru@unex.es)



Instituto Politécnico de Guarda (IPG,  
Portugal)

Contacto:  
Carolina Vila-Cha  
[cvilacha@ipg.pt](mailto:cvilacha@ipg.pt)

**Listado de socios**

# ÍNDICE

## RESUMEN EJECUTIVO

1. **CORTEX+ y Agentes de la arquitectura**
2. **Navegación social de SAR**
3. **Actividades para la estimulación física-cognitiva**

[Bibliografía](#)

# RESUMEN EJECUTIVO

---

El presente documento describe brevemente la selección de desarrollos y aplicaciones desarrolladas a lo largo del proyecto EuroAGE para la interacción humano-robot. En primer lugar, se describe la arquitectura cognitiva CORTEX+ desarrollada en el marco del proyecto EuroAGE como evolución de la arquitectura CORTEX, definiendo la red de agentes software desarrollados. A continuación, se introducen las aplicaciones implementadas en el proyecto EuroAGE para la interacción humano-robot por cada uno de los socios, entre las que destacan la navegación de un robot social asistencial, y la propuesta de actividades para la estimulación física, cognitiva y socio-emocional.

# 1. Desarrollos y aplicaciones para la Interacción Humano-robot

A lo largo del proyecto EuroAGE los distintos socios han desarrollado agente software para diferentes fines. Dar funcionalidad a un robot y a un entorno de asistencia ALab no es una tarea sencilla, y son necesarios múltiples agentes software que funcionen de manera coordinada y distribuida. Las arquitecturas desarrolladas en el proyecto, entre ellas CORTEX+, favorece la integración de los agentes software, facilitando el despliegue de funcionalidades en los robots sociales.

La interacción humano-robot es una de las actividades más importante que ha de tener desarrollada una plataforma robótica. Para llevar a cabo una correcta y socialmente aceptada HRI, es necesario disponer no sólo de algoritmos de generación de voz o de reconocimiento de voz o un gestor de las conversaciones, si no también acciones tan triviales como detectar la posición de la persona, el rostro de la misma, la navegación hacia la persona, etc.

En el proyecto EuroAGE se han desarrollado diferentes aplicaciones software que dan funcionalidad a las tareas anteriormente descritas. La Fig. 1 muestra un ejemplo de varios agentes software desarrollados para permitir una navegación de un lugar a otro de destino con el objetivo de avisar de un cambio en las actividades según la agenda del Centro.

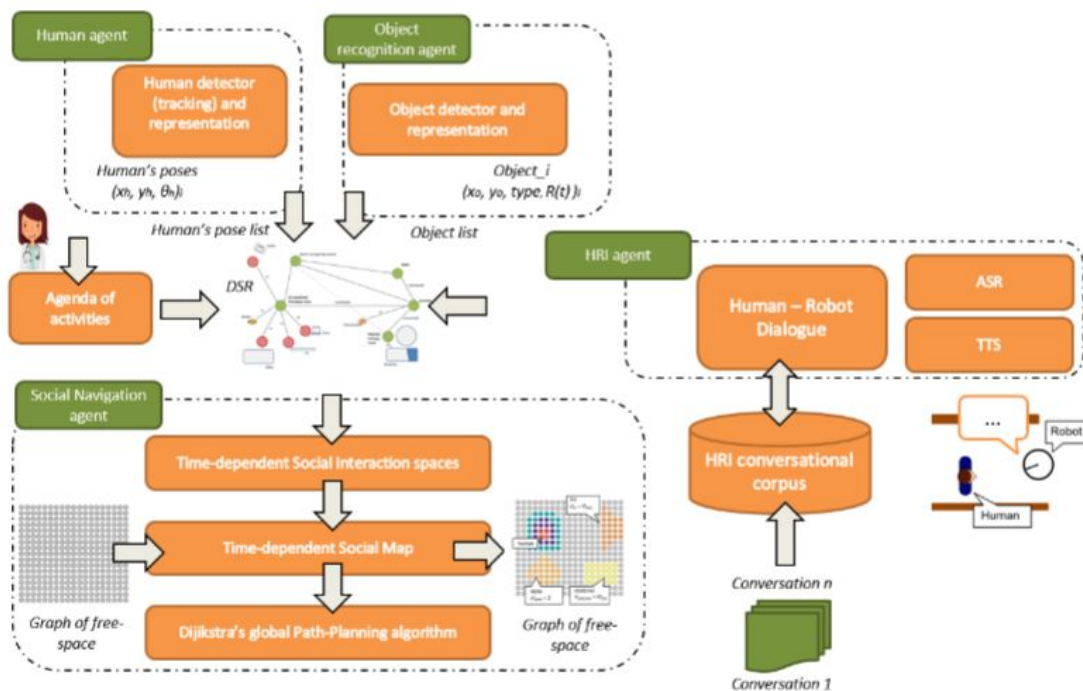


Figura 1. Ejemplo de desarrollos software en el marco del proyecto EuroAGE: robot que avisa de un cambio en la actividad programada en la agenda del centro.

Como se aprecia en la figura, el simple hecho de interactuar con una persona para indicarle que la actividad llega a su fin, implica varias acciones que el robot debe llevar a cabo: 1) navegar, lo que implica localizarse, construir un mapa, evitar obstáculos, detectar personas, detectar objetos, calcular los espacios personales para intentar no invadirlos; 2) planificar las acciones consecutivas que el robot debe ir ejecutando; y 3) gestionar la interacción humano-robot en sí misma. De la imagen anterior se desprende la complejidad de cualquier desarrollo de aplicaciones a lo largo del proyecto EuroAGE.

Para facilitar este enorme despliegue de recursos software, en el proyecto EuroAGE se amplió la arquitectura robótica cognitiva CORTEX+. Esta arquitectura se basa en una información compartida (DSR, Deep State Representation) por los diferentes agentes que interactúan con el entorno. Estos agentes leen y modifican el grafo según sus necesidades, manteniendo siempre la coherencia de los datos almacenados. La Figura 2 muestra un esquema representativo de la arquitectura con los agentes, locales en el robot o distribuidos a lo largo de ALab, interactuando con el DSR.



Figura 2. Esquema representativo de la arquitectura CORTEX+. Los agentes, locales en el robot o distribuidos en el ALab leen y modifican el grafo para ejecutar sus acciones

El grupo RoboLab y sus socios del proyecto han creado varios repositorios para el seguimiento del software desarrollado a lo largo del proyecto EuroAGE. La tabla siguiente muestra estos repositorios y el número de componentes desarrollados a lo largo de la vida del proyecto.

Repositorio	Agentes	Descripción
<a href="https://github.com/robocomp/robocomp-viriato">https://github.com/robocomp/robocomp-viriato</a>	<ul style="list-style-type: none"> <li>HumanSceneSim</li> <li>SocialNavigationAgent</li> <li>SocialNavigationGUI</li> <li>aggtFileCreatorAgent</li> <li>conversationalAgent</li> <li>faceDetector</li> <li>faceTracking</li> <li>fakeHumanAgent</li> <li>humanAgent</li> <li>humanObserverAgent</li> <li>humanObserverAgent_GUI</li> <li>humanPose</li> <li>humanPoseFake</li> <li>keyboardrobotcontrollerOmni</li> <li>localizationAgent</li> <li>localizationAgent_RCIS</li> <li>newSocialNavigationAgent</li> <li>roomLocalizationAgent</li> <li>socialRulesAgent</li> <li>socialnavigationGaussian</li> <li>viriatobase</li> </ul>	<p>Repositorio del proyecto EuroAGE para el robot Viriato. Agentes de la arquitectura para navegación, percepción de personas y objetos, interacción humano-robot, base, etc.</p>
<a href="https://github.com/robocomp/robotherapy">https://github.com/robocomp/robotherapy</a>	<ul style="list-style-type: none"> <li>bbdd</li> <li>robotTherapy</li> <li>therapyAnalysis</li> <li>therapyManager</li> </ul>	<p>Repositorio para la gestión de actividades y terapias propuestos por el robot</p>
<a href="https://github.com/robocomp/eur-oage-tv">https://github.com/robocomp/eur-oage-tv</a>	<ul style="list-style-type: none"> <li>adminGame</li> <li>bbdd</li> <li>fakeTV</li> <li>gameManager</li> <li>integratedHand</li> <li>tvGames</li> <li>tvGamesAdmin</li> </ul>	<p>Repositorio para la gestión y terapias propuestos por el ALab</p>
<a href="https://github.com/ljmanso/AGM">https://github.com/ljmanso/AGM</a>	<ul style="list-style-type: none"> <li>AGGLPlanner</li> <li>doc</li> <li>docker/bionic</li> <li>examples</li> <li>libagm</li> <li>parser</li> <li>python</li> <li>share/icons</li> <li>tmp</li> <li>tools</li> </ul>	<p>Repositorio del core de CORTEX+ y del DSR.</p>



## 2. Navegación social de SAR

---

La navegación social es un tema de enorme interés en la robótica autónoma. Los robots en escenarios con humanos tienen que ser capaces de comportarse de una manera socialmente aceptable, es decir, de planear un camino y navegar de acuerdo a reglas sociales, por ejemplo, los robots deben evitar acercarse a las personas o molestar a las personas que no están dispuestas a interactuar con ellas. Esta sección describe dos escenarios experimentales evaluados en el marco del proyecto EuroAGE.

### Framework de navegación social

Para calcular estos caminos socialmente aceptados es necesario construir un mapa social del medio. Por esta razón, el ALab necesita: i) detectar y modelar las personas (posición y orientación); ii) modelar su espacio social de interacción; iii) agrupar a las personas en caso de que se involucren en la interacción, modelando el espacio que necesitan para hacer dicha interacción; y iv) detectar los objetos y modelar su espacio de interacción. Además, para evitar bloqueos durante la navegación también es necesario proporcionar al robot la capacidad de seleccionar diálogos específicos mientras navega. La Fig. 1 mostraba un diagrama del framework que se plantea en el proyecto RoboAGE<sup>1</sup>.

En primer lugar, se parte de la creación de un mapa social basado en los espacios de interacción. Generalmente, las personas no quieren ser molestadas mientras interactúan con otras personas u objetos. En los centros de atención son comunes las terapias físicas o cognitivas en las que las personas - de edad avanzada o profesionales - interactúan con objetos o personas. Los robots deben ser capaces de detectar estas situaciones antes de planear su camino. En este sentido, la literatura define el concepto de Espacios de Affordances para referirse a las regiones en las que los seres humanos suelen realizar actividades específicas [1] (por ejemplo, las áreas cercanas a una mesa durante la terapia en ella). Estos espacios se denominan Espacios de Actividad cuando las personas interactúan con ellos. La construcción del mapa social del centro de cuidados es uno de los principales objetivos de este marco de navegación. Para lograrlo, el ALab primero detecta la posición de las personas en el mundo desde la red de cámaras RGBD. A partir de la imagen de profundidad, el agente de detección y seguimiento de los seres humanos proporciona la posición de las personas en las coordenadas

---

<sup>1</sup> Podemos encontrar más información relacionada con este framework en el Entregable E.2.3

esperadas. Algo similar hace el agente encargado de detectar objetos en el centro de cuidado. Una vez que el modelo DSR ha sido actualizado, se crean los modelos de espacio personal, y los modelos para interacción, tal y como se definen en nuestro trabajo publicado en [2]. Así, se describen los espacios personales de las personas como gaussianas asimétricas, y los espacios de los objetos, diferentes formas según el tipo de objeto con el que la persona interactúa. La Fig. 3 muestra la forma del DSR al inicio del movimiento en el escenario descrito

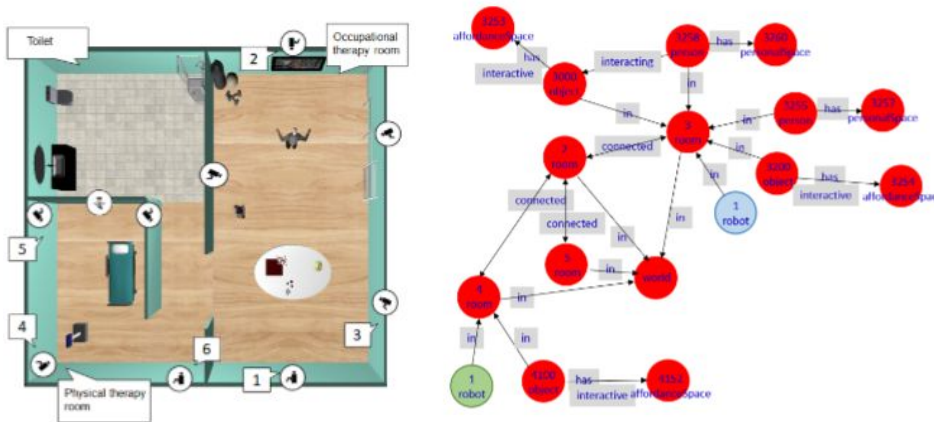


Figura 3. Vista en 2D del escenario simulado, con los sensores RGBD; vista del DSR asociado a ese mundo simulado.

Una vez descrito los espacios de interacción, se modifican los pesos del mapa de coste, un grafo de espacio libre que almacena la información de los obstáculos del entorno. Esta modificación de coste se hace en función de los espacios de interacción [3]. Finalmente, sobre el grafo de espacio libre final -el mapa social-, se establece una ruta óptima utilizando el algoritmo de Dijkstra. La Fig. 4 muestra los resultados de una simulación en el entorno presentando a la izquierda.

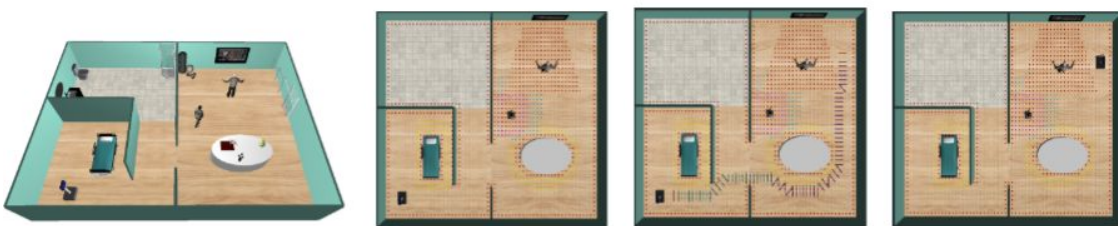


Figura 4. Vista en 3D del escenario simulado; espacios de interacción social, tanto para las personas (el anciano y el cuidador) como para los objetos en el ALab; trayectoria planeada por el robot en el ALab; y la pose del SAR al final del caso de uso



## 3. Actividades para la estimulación

### física-cognitiva

---

Para poder llevar a cabo políticas de envejecimiento activo, es necesario garantizar una adecuada protección, seguridad y cuidado de las personas. Eso incluye tratamientos personalizados, seguimiento a largo plazo que incluya una evaluación continua de las capacidades de la persona, y la inclusión de tecnologías de monitorización, comunicación y terapia, tanto en el hogar como en hospitales, residencias o centros de día (en adelante, residencias). Estas tecnologías para el envejecimiento activo, englobadas bajo el término Ambient Assisted Living (AAL) [3], surgen para contribuir a la independencia de las personas mayores, pero también para aliviar la carga de trabajo de los profesionales sanitarios y cuidadores, sin reemplazarlos en ningún caso.

Los robots de asistencia social para el uso en el hogar y residencias están diseñados para proporcionar servicios a los usuarios a través de la interacción directa, como el apoyo a la comunicación con otras personas, mostrar información, entretener o promover actividades a los usuarios. Estas tareas han recibido un aumento considerable de atención en los últimos años, y por lo tanto, hay varios robots de asistencia disponibles en el mercado o con el objetivo de estar en el futuro cercano. Algunos de ellos son plataformas de peso ligero que requieren control manual ya que proporcionan principalmente la funcionalidad de telepresencia (por ejemplo, la plataforma Giraff dentro de los proyectos de ExCITE, o los Proyectos Europeos VictoryaHome o GiraffPlus), pero muchas otras plataformas son capaces de proporcionar ayuda autónoma.

Cuando el robot no se adapta al usuario, es visto como una herramienta. La adaptación permanente es una de las cuestiones clave para lograr que el usuario atribuya características de compañerismo al robot. Esta necesidad también se exige en esos robots que sólo brindan ayuda física, pero no social o ayuda - médica o de cuidados - especializada. La entrega final de los proyectos como Accompany o Mobiserv ponen el énfasis en esta cuestión. Debido a este hecho, los proyectos recientes como GrowMeUp, de uno de los socios del proyecto, dan un trato especial a los vínculos sociales y la prestación de atención, estando entre sus objetivos el desarrollo de una red de cuidados colaborativos. La capacidad de proporcionar compañía social y cuidados se debe complementar con la necesidad de dotar al robot con una personalidad específica. Es decir, es importante que el robot pueda dotar a las personas con servicios de bienestar para la estimulación social y cognitiva y el control general del estado - nutrición, salud -, pero también es obligatorio que el robot sea aceptado como un compañero real. La personalidad del robot y aceptación a largo plazo siguen siendo aspectos obligatorios que no todos los proyectos cubren. Los estudios muestran que el efecto novedad se pierde rápidamente y, después de eso, los usuarios pierden interés, cambiando su actitud hacia el robot [4]. No podemos olvidar, sin embargo, que es muy humano personalizar a nuestros animales de compañía y a las cosas cotidianas y debemos por tanto potenciar esa antrópica condición.

## Sistema de estimulación física mediante Avatar

Para cumplir con la funcionalidad requerida para ayudar a los usuarios ancianos en su vida diaria, se ideó la arquitectura para la Planificación y Ejecución de Multi-Agentes que se muestra en la Figura 5. Se ha optado por una arquitectura centralizada, con un planificador central responsable de tomar las decisiones de alto nivel dentro del sistema, y un número de agentes de ejecución distribuidos, responsables de ejecutar el plan de alto nivel. Este paradigma es el que mejor se ajusta al problema en cuestión: la naturaleza distribuida de la funcionalidad del sistema requiere un sistema de planificación de múltiples agentes, y la suposición de una conectividad total dentro de la casa inteligente permite un enfoque centralizado y monolítico.

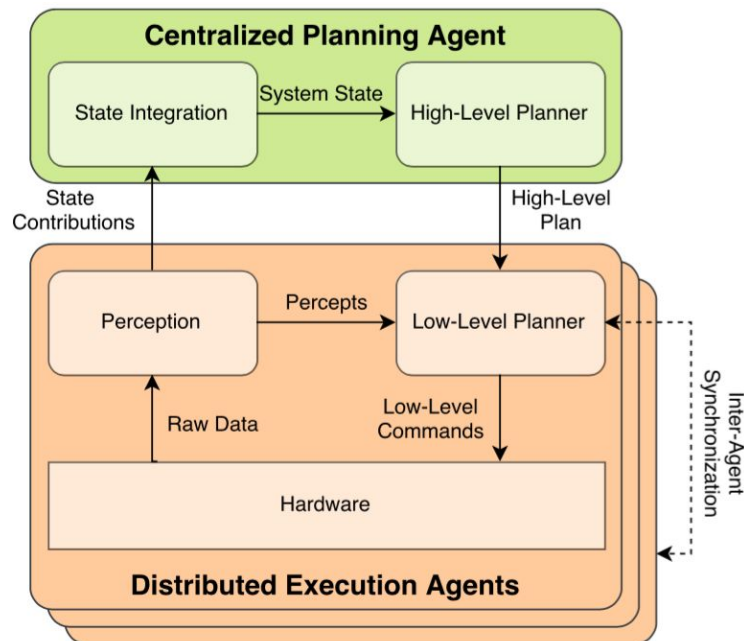


Fig. 5. Arquitectura de agentes en el sistema de estimulación física

La planificación centralizada requiere una visión unificada del estado del sistema en cada momento. Sin embargo, diferentes agentes pueden percibir la misma realidad de manera diferente, por ejemplo, un robot móvil y una red de cámaras pueden tener estimaciones diferentes de la posición del robot en el mundo. Cada agente puede contribuir potencialmente al estado del sistema enviando contribuciones estatales, que representan la perspectiva del agente sobre el estado del sistema, al sistema centralizado. Por lo tanto, se necesita un paso de integración de estados para filtrar y fusionar las perspectivas recibidas por los distintos agentes.

El estado del sistema resultante incluye toda la información necesaria para la correcta determinación de un nuevo plan de alto nivel. Por ejemplo, puede contener las posiciones de cada agente en el espacio, sus capacidades, su tarea actual, planes anteriores, etc. El tiempo en sí mismo es un elemento central en la integración de los estados, y proporciona información útil para el estado del sistema, por ejemplo, el hecho de que un agente haya dejado de informar sobre las contribuciones previstas durante cierto tiempo puede indicar un

estado de fracaso.

El estado del sistema integrado sirve de entrada a un planificador centralizado, que asigna los agentes a las tareas mediante una técnica de planificación automatizada. Como enfoque preliminar, empleamos una técnica basada en reglas que se ajusta a nuestros casos de uso, pero se pueden utilizar técnicas de planificación más sofisticadas. El plan de alto nivel consiste en la asignación de tareas a los agentes, que el sistema centralizado produce como respuesta a determinados estados del sistema. Un ejemplo de estos comandos sería la asignación del agente 0 al servicio de ejercicio de tareas. Los comandos de alto nivel son traducidos por los agentes en comandos de bajo nivel específicos para cada agente, tales como comandos de velocidad, salida de voz, salida visual, etc., que utilizan para cumplir su tarea asignada. Una coordinación más estrecha entre los agentes se produce en base a cada servicio, según los planes locales que cada agente está ejecutando.

La arquitectura puede acomodar dos tipos de agentes de ejecución: Agentes de percepción, que sirven como recolectores de datos para el sistema; Agentes situados, capaces de percibir el mundo pero también capaces de actuar o interactuar con los usuarios, y/o tomar decisiones de bajo nivel.

CORTEX es el principal facilitador de la comunicación para todo el sistema, llevando toda la comunicación entre los agentes y la comunicación entre los agentes y el sistema centralizado. Esto permite una integración sin problemas de los sistemas que funcionan en diversas topologías de red, así como una extensibilidad dinámica del sistema para dar cabida a un mayor número de agentes.

El segundo diagrama, presentado en la Fig. 6, muestra una instancia de la arquitectura general para abordar un caso de uso de promoción de ejercicio físico. Este caso hace uso de tres agentes: una red de cámaras, un avatar virtual y un robot social, es decir, un agente de percepción y dos agentes situados, respectivamente. Estos tres agentes actúan en tándem, compartiendo información y decisiones a través de la arquitectura presentada anteriormente, para llevar al usuario a realizar actividades físicas.

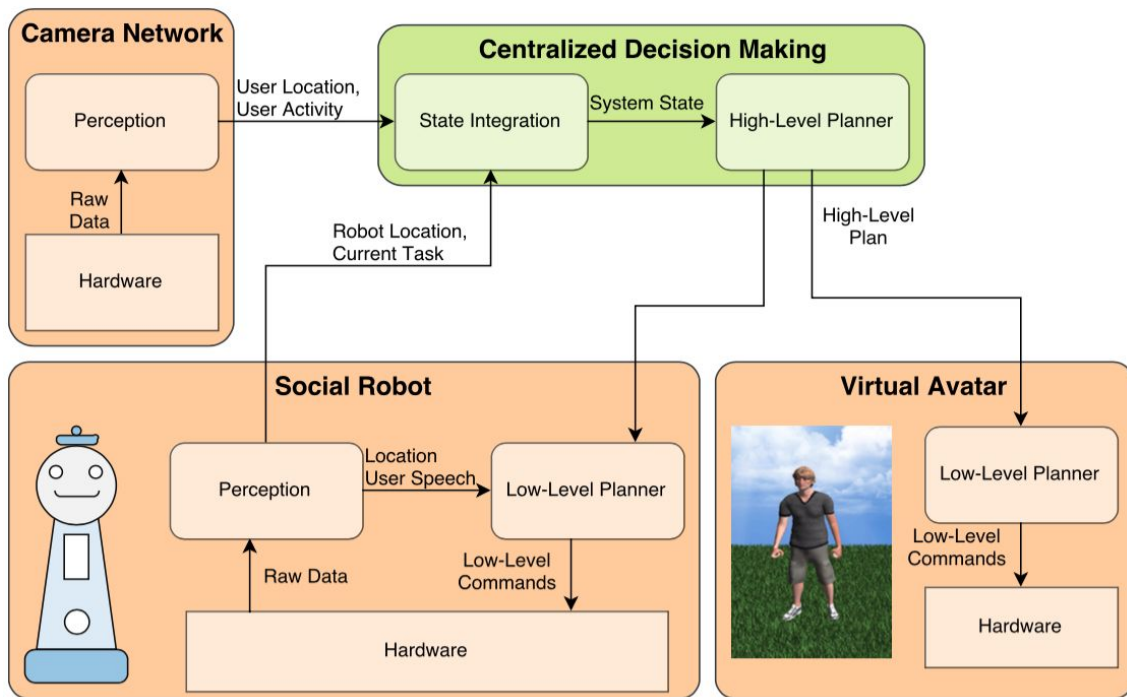


Fig. 6. Vista general de los agentes que llevan a cabo la actividad

Cada uno de los tres agentes contribuye al estado del sistema: el robot y el avatar informan al sistema de su tarea actual, y el sistema de cámaras sobre el estado del usuario. El integrador de estado fusiona esta información en un estado de sistema unificado, de modo que puede ser utilizada para la planificación. Si el sistema de cámaras detecta que el usuario está inactivo durante un tiempo preestablecido, el sistema de planificación centralizada asigna una tarea al avatar y al robot, activando la ejecución adecuada en ambos agentes.

La ejecución se lleva a cabo como se muestra en la Figura 7, con cada agente operando de forma independiente con puntos de sincronización bien definidos que permiten a los agentes cooperar durante la ejecución de la tarea. La información se comparte en ambos sentidos, con el robot recogiendo los parámetros para la rutina de ejercicio que se va a ejecutar y el avatar virtual proporcionando al robot la rutina a ejecutar. Esta implementación permite explotar las mejores capacidades de ambos agentes para realizar mejor la tarea en cuestión, por ejemplo, el robot social puede hacer uso de sus capacidades de movilidad y habla para movilizar y animar al usuario, y el avatar virtual puede explotar su alta precisión en la reproducción del propio ejercicio para actuar como instructor del usuario.

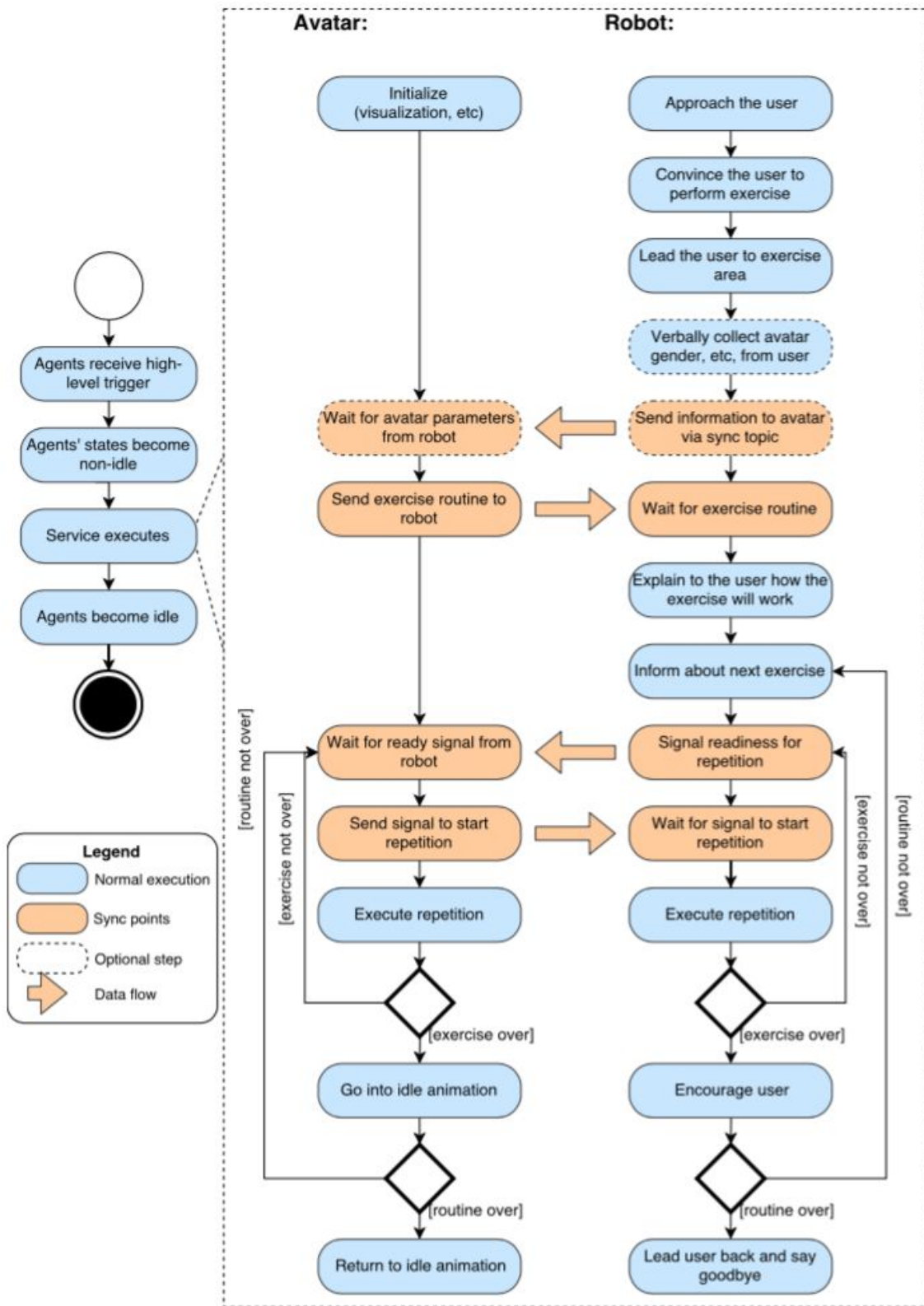


Figura 7. Integración de la interacción humano-robot



## Sistema de estimulación cognitiva y socio-emocional

El siguiente módulo permite al robot (o al personal sanitario) proponer actividades lúdico/informativas a los ancianos. Para ello el robot debe, en general, conocer la posición de la persona en el mundo, la agenda de actividades del centro y la asignada a cada usuario. A continuación, o bien de forma automática o bien de forma supervisada por el profesional, se debe proponer una actividad cognitiva o informar de actividades a la persona mayor:

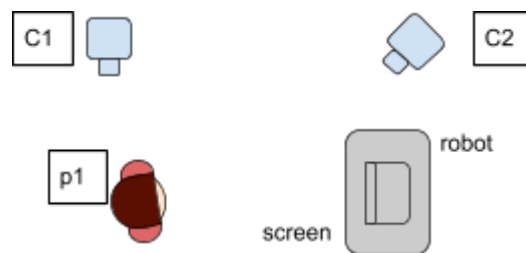


Figura 9. Entorno de trabajo: el robot le propone juegos o información al anciano de forma automática o bien a través de supervisión.

Conocida la  $p_i$  que representa a una persona en el mundo, el robot debe navegar socialmente hasta donde se encuentra la persona y en ese momento proponerle la actividad.

Los juegos/terapias tienen un patrón común, que se resume a continuación de forma esquemática:

- a. **Entrada: descripción de la persona  $p_1$  (pose, esqueleto, datos sociales/estado, evolución métricas)**
- b. **Manejo de DialogFlow (o Rasa chatbot) para la gestión de la conversación con las personas en el entorno del robot.**
- c. **Gestión del juego por parte del profesional**

### a. Descripción de la persona

Una persona  $id_i$  en el proyecto EuroAGE se caracteriza en el DSR por los siguientes atributos:

1. Nombre
2. Tipo (paciente/terapeuta)
3. Pose,  $p_i$
4. Tiempo en reposo/activo
5. Actividad que realiza en un instante
6. Tiempo sin actividad cognitiva/física/emocional

Conocida la  $p_i$  que representa a una persona en el mundo, el robot debe navegar socialmente hasta donde se encuentra la persona y en ese momento proponerle la actividad. Los algoritmos de navegación utilizados están publicados en [3] y [4], y descritos en este documento.

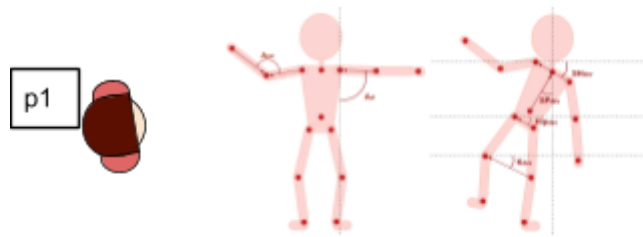


Figura 8. Una persona - anciano o terapeuta - tendrá que tener una ubicación en el mundo, así como otras características extraídas, en su mayoría, del análisis del cuerpo.

## b. Gestión del diálogo hri

DialogFlow es una herramienta de Google desarrollada para generar chatbot con lenguaje natural. DialogFlow se usa en este sistema para llevar a cabo el flujo del diálogo de una terapia entre el humano y el robot.

En las definiciones comunes de un chatbot encontramos los siguientes términos, particularizando para nuestro sistema concreto:

- a) Agentes: en este caso será el módulo, sistema en general. Será denominado de aquí en adelante como Agente guía durante la terapia.
- b) Intents: se denomina con este término a las intenciones del anciano. En la definición del Agente tenemos que tener en cuenta cuáles serán las preguntas/peticiones/contestaciones del anciano durante el desarrollo de la terapia.
- c) Entities: Permite extraer parámetros o variables de la conversación con el anciano.
- d) Actions: Se denomina así a las acciones que llevará a cabo el robot/terapia tras detectar un Intent (una nueva conversación o lanzar un mensaje al terapeuta, etc)
- e) Contexts: Permite definir intenciones (Intents) en función del contexto de la comunicación, esto es, de las conversaciones anteriores.

Con todo, las terapias y las actividades propuestas deben estar desarrolladas para que todas o casi todas siguieran el mismo patrón generado en dialogFlow. A continuación mostramos un ejemplo de conversación según dialogFlow para una terapia concreta:

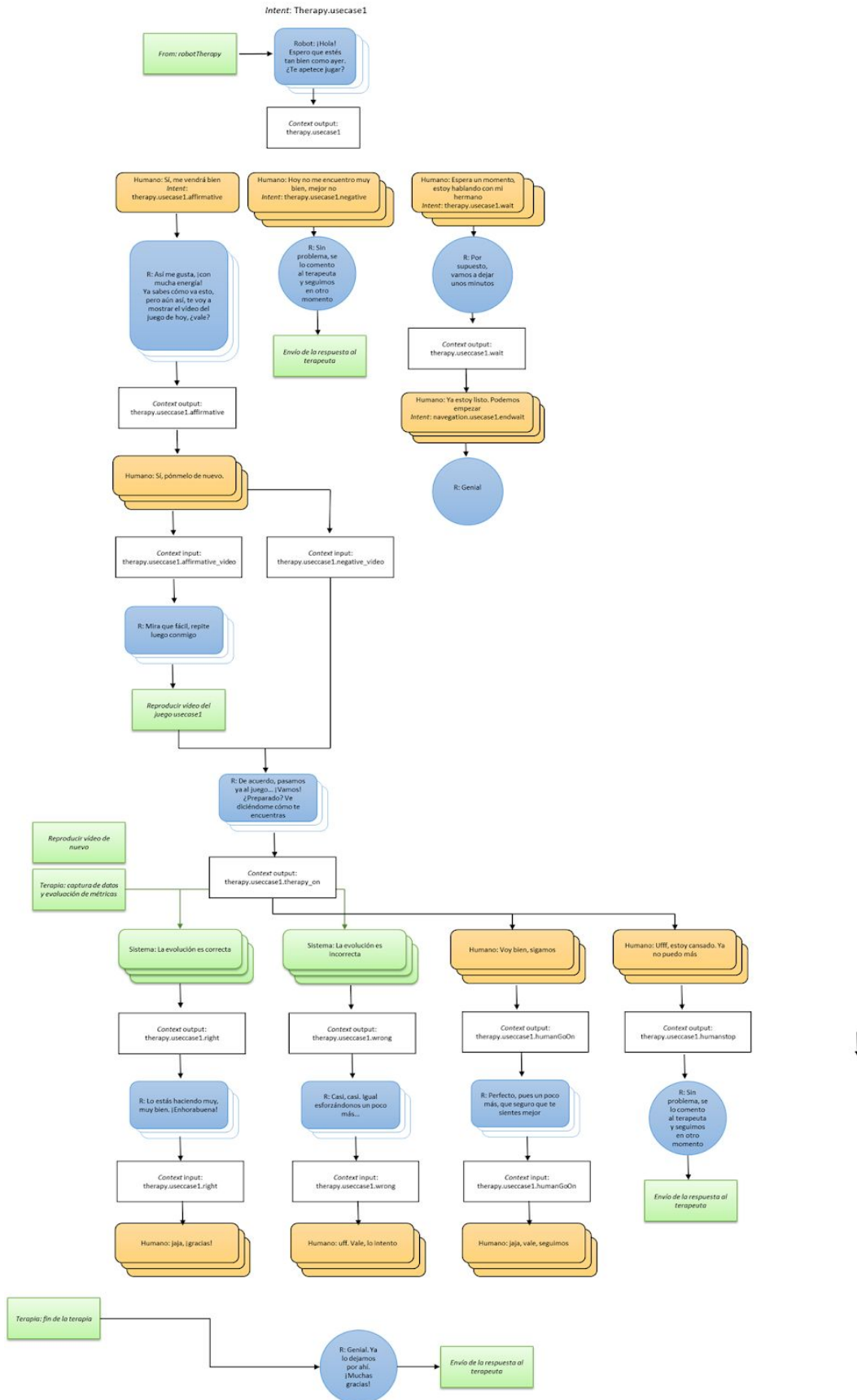


Figura 9. Gestión del diálogo durante la terapia

### c. Gestión del juego por parte del profesional

La Figura 10 muestra un esquema del sistema propuesto para proponer y dirigir las terapias a los pacientes. Una descripción más detallada se hace a continuación:

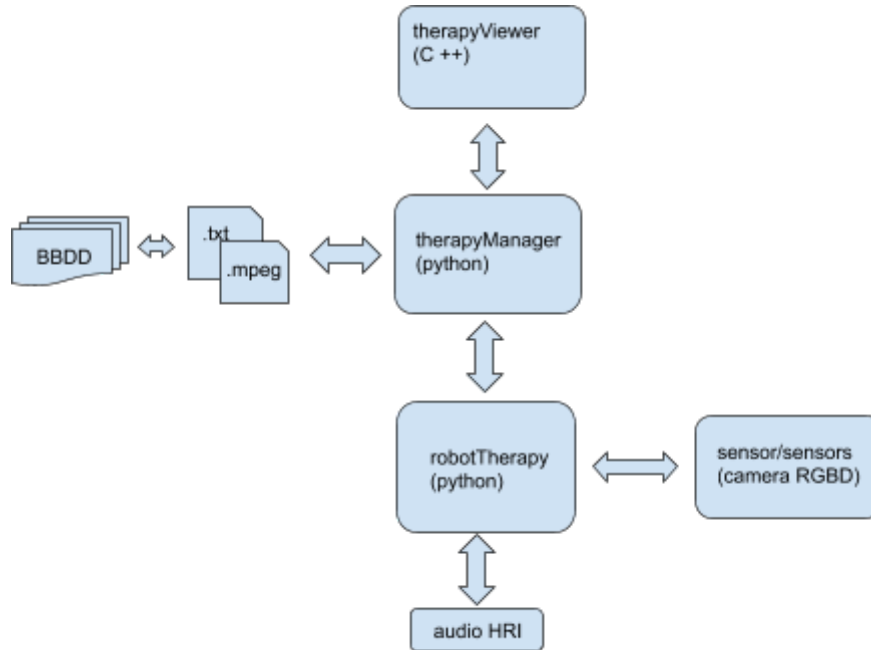


Figura 10. Diagrama propuesto para las terapias del proyecto EuroAGE

### Persona mayor

La interacción del anciano con el sistema se realiza únicamente a través de las sesiones de terapia planificadas por el terapeuta.

### Terapeuta

La interacción del terapeuta con el sistema se podría realizar con un dispositivo fijo, PC conectado en local, y/o con un dispositivo móvil, Tablet, a través de una versión simplificada web. Las funcionalidades no serán exactamente las mismas, estando la versión fija más desarrollada con la posibilidad de visualizar offline el ejercicio realizado por el anciano.

El terapeuta busca la siguiente funcionalidad inicial en el sistema:

- a) Acceder a los datos del anciano. Entre otra información, aparte de la propia personal, interesa conocer cómo evoluciona durante las terapias.
- b) Proponer juegos a los ancianos
- c) Visualizar/reproducir una sesión de terapia en concreto
- d)

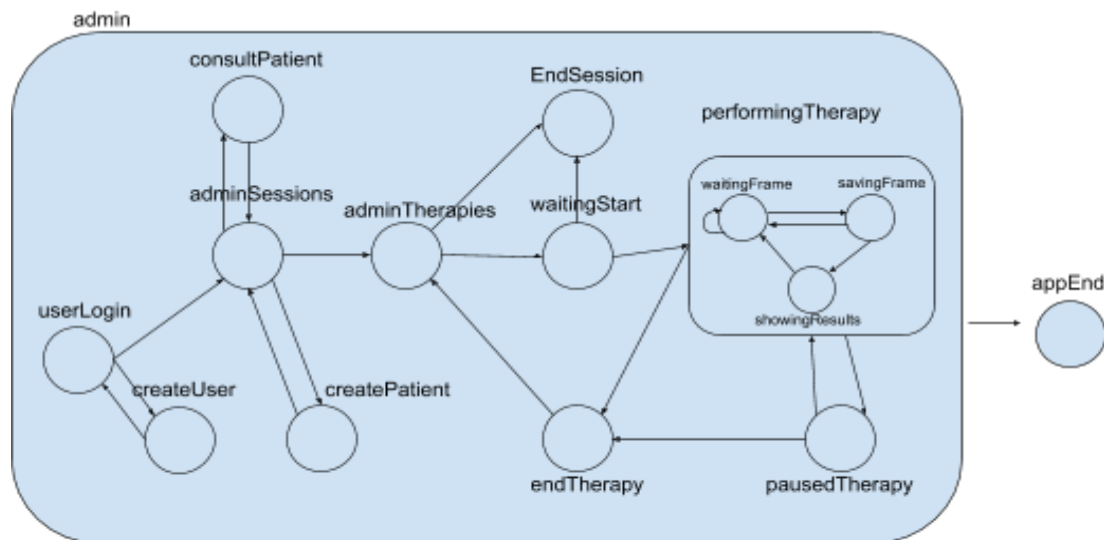
Para cumplir estas especificaciones, definimos los siguientes módulos:

## therapyManager

El componente therapyManager es el interfaz del terapeuta a las terapias realizadas - o a realizar - por el anciano. Los **objetivos funcionales** son los que describimos a continuación:

- El terapeuta debe:
  - Iniciar sesión (login)
  - Crear nuevo usuario (nuevo terapeuta)
  - Seleccionar paciente (guardados en BBDD)
  - Crear nuevo paciente
  - Seleccionar actividad física/cognitiva (terapia a realizar)
  - Visualización de los resultados e histórico de un paciente (métricas a definir que estarán almacenadas en la BBDD y que precisan realizar consultas a la misma)
  - Generar informes - pdf.
  - Visualizar/reproducir una terapia (lanzando el componente necesario<sup>2</sup>)

La máquina de estado que define el componente therapyManager se representa a continuación:



User: Terapeuta

Patient: Anciano  
Therapy: Terapia a  
realizar/consultar

Al seleccionar al player se debe dar doble funcionalidad: poder lanzar una nueva terapia y poder consultar sus datos. Las consultas pueden ser tan complejas como se desee al final del proyecto. Inicialmente pueden ser básicas

<sup>2</sup> Cuando el terapeuta inicia una terapia o desea visualizar los datos almacenados de una sesión pasada se accede al componente therapyViewer (Sólo podría lanzarse desde un PC)

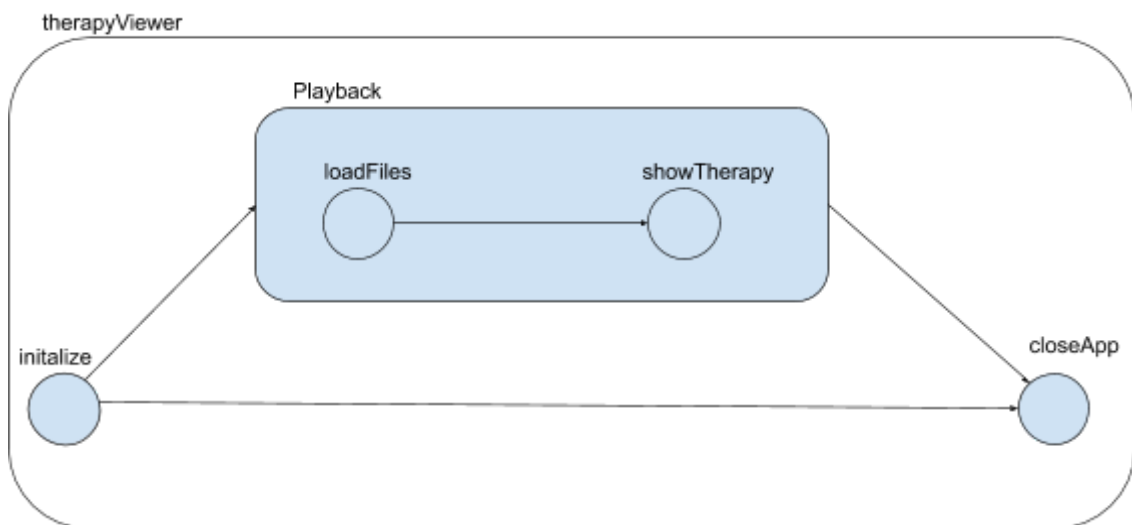
## therapyViewer

Se trata de un componente de visualización y tratamiento de los datos. Este componente es llamado desde therapyManager con información sobre el usuario y la sesión en concreto. Es una sesión guardada en BBDD, que permite al terapeuta la reproducción de un ejercicio para su estudio en detenimiento.

Ofrece varias funcionalidades:

- El componente mostrará el video y la información del esqueleto obtenida para dicha actividad.
- Permitirá visualizar en forma de tabla las métricas calculadas durante toda la sesión.
- Permitirá, en la medida de lo posible, controlar la reproducción - velocidad, retroceso...

La máquina de estados para el componente TherapyViewer puede observarse a continuación:



**robotTherapy:** Este componente será quien realice la terapia definida por los Terapeutas ocupacionales. Dependiendo de la terapia en cuestión, se realizará un proceso u otro con los sensores necesarios. Como **elementos de entrada** - llamados desde el therapyManager tenemos que intentar que sean comunes a todos los juegos/terapias: nombre del paciente, datos de la sesión(es) anteriores, id de la terapia... Cada terapia tendrá su definición y elementos característicos, métricas, diálogo...

) | k MI ? nUn { ? } ZxM { ? } lnl ' .

) | k MI ? nUn { ? } ZxM { ? } lnl ' .

El componente debe tener interacciones con el usuario (dialogFlow) del tipo:

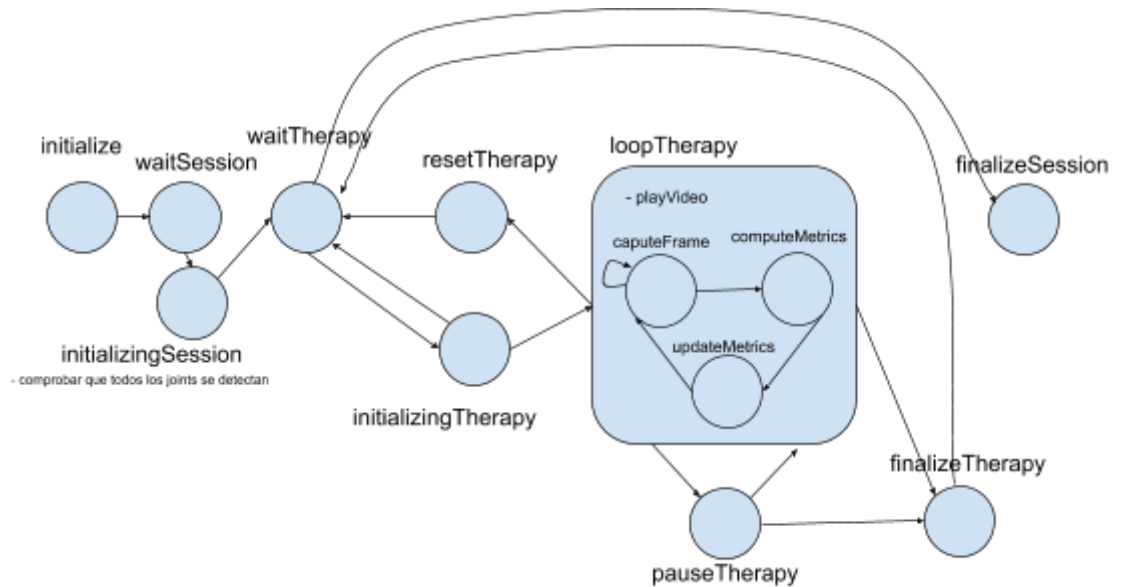
- a) saludos iniciales
- b) explicación del juego
- c) frases de interacción durante el desarrollo del juego
- d) despedida

Del mismo modo debe reproducir vídeo (explicación de las terapias) al principio de la misma.

La funcionalidad concreta de cada terapia será definida previamente. Durante la terapia y al final de la misma, se debe:

- a) Almacena los datos captados por el sensor del robot (video, joints, métricas).
- b) Analiza los datos, calculando las métricas y mostrándolas al terapeuta en tiempo real.
- c) Interaccionar con el usuario

La máquina de estado quedaría:



Cada actividad/sesión será diferente, las métricas variarán de una a otra. Será necesario establecer qué métricas mostrar en cada actividad, así como el rango de valores en los que se tienen que encontrar las métricas calculadas.

Los resultados (**salida**) de la sesión (texto, vídeos, audios, id del tipo de terapia (una unívoca que identifique a este juego en concreto)...) serán almacenados en la BBDD. En la BBDD se almacenan las métricas que se describan en la definición de las terapias.



# Bibliografía

---

[1] Rios-Martinez, J. Socially-aware robot navigation: combining risk assessment and social conventions. Hal. Inria France, 2013

[2] L. V. Calderita, A. Vega, S. Barroso-Ramírez, P. Bustos, and P. Núñez. Designing a Cyber-Physical System for Ambient Assisted Living: a Use Case for Social Robot Navigation in Caregiving Centers, sensors 2020.

[3] J. Broekens, M. Heerink, and H. Rosendal. Assistive social robots in elderly care: A review. Gerontechnology, 8, 94–103, 2009.

[4] Leite I, Martinho C, Paiva A. Social robots for long-term interaction: a survey. Int. J. Soc. Robot, 2013