



Universidad
Zaragoza

Proyecto Fin de Carrera

Estudio y diseño de un sistema de
monitorización de actividades en hogares
con ancianos

Autor

José Luis Cáceres Franco

Director

Antonio Bono Nuez

Escuela de Ingeniería y Arquitectura
2011

A María, porque sin su apoyo y ayuda
no habría sido capaz de realizar
este proyecto, gracias.

A mis padres, por la posibilidad
de formación que me han dado.

A mi tutor y a Tecnodiscap
especialmente a R. Blasco
por su ayuda desinteresada.

RESUMEN

Debido al aumento de la esperanza de vida, se ha producido un incremento de la población anciana en las últimas décadas. Estas personas, con la edad suelen sufrir una pérdida de autonomía. Con la idea de poderles ayudar a vivir durante más tiempo en buenas condiciones en su hogar, surgió este proyecto que se ha desarrollado dentro del grupo de investigación Tecnodiscap.

Para conseguir este objetivo, se ha necesitado disponer de una herramienta que pudiese detectar si la persona por sí misma era capaz de realizar las actividades de la vida diaria. Para eso, se ha buscado información de otros autores sobre la forma en que conseguían detectar actividad en los hogares y se ha creado un prototipo de la herramienta. Se ha desarrollado un Script en Matlab de recogida de datos y se ha diseñado una ubicación adecuada de los sensores en el hogar. Tras esto, se ha realizado una recogida de datos y un Script de tratamiento de esos datos. Posteriormente, se han traducido estos datos a actividades. Se ha validado la herramienta comparando los resultados con las actividades que el usuario había registrado manualmente.

En los resultados se ha observado que el reconocimiento de actividades con la herramienta desarrollada ha coincidido en gran medida con las registradas por el usuario.

Como conclusión, se puede decir que se ha conseguido diseñar una herramienta lo suficientemente robusta, económica, sencilla y no invasiva que puede ser instalada fácilmente en los hogares sin necesidad de obras.

ÍNDICE

CAPÍTULO 1: INTRODUCCIÓN.....	- 10 -
1.1 Marco del proyecto:	- 10 -
1.2 Motivación del proyecto:	- 10 -
1.3 Estructura de la memoria:	- 11 -
CAPÍTULO 2: OBJETIVOS.....	- 13 -
2.1 Especificaciones del proyecto	- 13 -
2.2 Especificaciones técnicas:	- 14 -
2.3 Distribución temporal del desarrollo del proyecto	- 15 -
CAPÍTULO 3: ESTUDIO DEL ESTADO ACTUAL DEL ARTE	- 16 -
CAPÍTULO 4: DESARROLLO DEL PROYECTO	- 26 -
4.1 Diseño de aplicaciones:	- 26 -
4.1.1. Introducción	- 26 -
4.1.2. Diagrama de bloques	- 27 -
4.1.3. Herramientas globales	- 28 -
4.1.3.1. Matlab:	- 29 -
4.1.3.2. Telegesis Terminal:	- 32 -
4.1.3.3. Instalación drivers:	- 36 -
4.2 Hardware:	- 37 -
4.2.1. Plogg:	- 37 -
4.2.2. Multisensores:	- 39 -
4.2.3. Comunicación Zigbee:	- 40 -
4.2.3.1. Introducción	- 40 -
4.2.3.2. Estándar IEEE 802.15.4	- 40 -

4.2.3.3.	Tipos de dispositivos	42 -
4.2.3.4.	Topologías y funcionamiento de la red	43 -
4.2.3.5.	Comunicaciones	44 -
4.2.3.6.	Mensajes de datos.....	46 -
4.2.3.7.	Seguridad.....	46 -
4.2.3.8.	Perfiles.....	46 -
4.2.4.	Instalación de los sensores:	47 -
4.3	Software:	52 -
4.3.1.	Script de recogida de datos:.....	52 -
4.3.2.	Desarrollo del Script de los Ploggs:	53 -
4.3.2.1.	Funcionamiento del programa:.....	55 -
4.3.2.2.	Diagrama de Flujo del Script de recogida de datos de los Ploggs:.....	61 -
4.3.2.3.	Dificultades añadidas durante el desarrollo de la aplicación:.....	63 -
4.3.3.	Script de análisis de datos:	64 -
4.3.3.1.	Diagrama de Flujo del Script de tratamiento de datos del multisensor: ..	65 -
4.3.3.2.	Diagramas de Flujo del Script de tratamiento de datos de los Ploggs:.....	66 -
4.4	Proceso de recogida de datos en el experimento:.....	68 -
4.5	Actividades reconocidas:.....	72 -
4.5.1.	Higiene:	72 -
4.5.2.	Comida:	76 -
4.5.3.	Ocio:	82 -
4.5.4.	Entrada/Salida de casa:.....	85 -
4.5.5.	Ir al baño:	87 -
4.5.6.	Descanso:	90 -
4.5.7.	Un día completo:	93 -
CAPÍTULO 5: CONCLUSIONES.....		94 -
5.1	Conclusiones del trabajo:	94 -

5.2	Cumplimiento de objetivos:	- 95 -
5.3	Desarrollos futuros:	- 96 -
5.4	Conclusiones personales:	- 98 -
BIBLIOGRAFÍA.....		- 99 -
ÍNDICE DE FIGURAS.....		- 102 -
ÍNDICE DE GRÁFICOS		- 104 -
ÍNDICE DE TABLAS		- 106 -
ANEXO 1: CÓDIGO DEL PROGRAMA DE RECOGIDA DE DATOS		
ANEXO 2: CÓDIGO DEL PROGRAMA DE ANÁLISIS DE DATOS		
ANEXO 3: ARTÍCULOS		
ANEXO 4: PROTOCOLOS BINARIOS DE LOS PLOGGS		
ANEXO 5: COMANDOS TELEGENESIS TG-ETRX-R212-AT		

CAPÍTULO 1:

Introducción

1.1 Marco del proyecto:

Este proyecto se ha desarrollado dentro del grupo de investigación Tecnodiscap [1], enmarcado en el Instituto de Investigación en Ingeniería de Aragón (I3A) [2], de la Universidad de Zaragoza. El objetivo principal del grupo Tecnodiscap es la mejora de la calidad de vida de las personas con algún tipo de discapacidad, dependencia o necesidades especiales mediante servicios basados en la aplicación de las Tecnologías de la Información y Comunicaciones (TIC).

En concreto, mi trabajo se ha integrado dentro de la línea de investigación y colaboración que lleva el grupo en el ámbito de la dependencia de las personas mayores, siendo este proyecto parte de un conjunto de mejoras para la valoración de la calidad de vida de dichas personas.

1.2 Motivación del proyecto:

Actualmente, en nuestro país, las personas viven cada vez más años. Por este motivo, se ha producido un aumento de la población anciana en las últimas décadas.

Estas personas de edad avanzada, suelen querer envejecer en su hogar, lo que conlleva unos riesgos asociados por la pérdida de autonomía que sufren con el paso de los años. Por este motivo surgió la idea de este proyecto.

Para ayudarles a vivir durante más tiempo en buenas condiciones, se necesita una herramienta que permita detectar si la persona por sí misma es capaz de realizar las actividades cotidianas del hogar.

A día de hoy, para medir el grado de dependencia que tiene la persona se puede utilizar, entre otros, un procedimiento definido por Katz [3, 4, 5]. Este método, se basa en analizar el grado de realización de las actividades de la vida diaria. Normalmente, los datos se recogen mediante encuestas que efectúan los servicios sociales. El problema de esta metodología es que con la realización de encuestas se pueden generar resultados un tanto subjetivos que nos conllevan a errores. Esto es debido a que las respuestas obtenidas dependen mucho del estado anímico de la persona, de su manera de interpretar la pregunta o de que infravalore o sobrevalore la respuesta, especialmente si su deseo es envejecer en su hogar. Por lo tanto, los resultados obtenidos mediante esta herramienta no son todo lo objetivos que se desearían.

Para poder hacer que los datos sean lo más objetivos posibles y así ayudar a los servicios sociales, se ha intentado diseñar un programa sencillo, no invasivo, respetuoso con la vida privada de las personas y que funcione sin la necesidad de una participación activa de los ocupantes. Además, la recogida de datos ha sido de forma continua, por lo que se ha generado resultados durante todo el tiempo.

1.3 Estructura de la memoria:

La memoria se ha estructurado en 5 capítulos. Al final de los cuales se han añadido 5 anexos para completar los aspectos que en la memoria no se han explicado tan extensamente como se requiere.

El contenido de cada capítulo es el siguiente:

- En el primer capítulo, se ha comentado una visión general del proyecto situándolo en el contexto de trabajo.
- En el segundo capítulo, se han especificado los objetivos así como la planificación de trabajo.
- En el tercer capítulo, se ha hecho un estudio del estado actual del arte. Se ha recogido una búsqueda de las distintas soluciones que dan otros autores para el reconocimiento de actividad en el hogar.

- En el cuarto capítulo, se ha descrito el desarrollo de las aplicaciones que consta este proyecto. Este capítulo se ha dividido en cinco partes fundamentales. En la primera, se ha enfocado una introducción, unos diagramas de bloques generales y una descripción de las herramientas que se han empleado. En la segunda parte, se ha recogido el diseño de hardware del sistema. En la tercera se ha descrito el diseño del software, generando dos Scripts, uno de recogida y otro de tratamiento de los datos. En la cuarta parte, se han recogido los datos de forma continua en un piso de prueba durante un periodo de días consecutivos. Y en la quinta y última parte, se ha intentado encontrar actividades a partir de los datos obtenidos.
- En el quinto capítulo se ha realizado un análisis de los objetivos cumplidos, las conclusiones obtenidas y las líneas de trabajo futuras.

En cuanto a los anexos:

- En el primer anexo, se ha añadido el código del programa de recogida de datos de los Ploggs.
- En el segundo anexo, se ha recogido el código del programa de análisis de los datos.
- En el tercer anexo, se han recopilado los artículos más importantes mencionados en el estado del arte.
- En el cuarto anexo, se han adjuntado los protocolos binarios de los Ploggs.
- En el quinto anexo, se han adjuntado los comandos de Telegesis versión TG-ETRX-R212-AT.

CAPÍTULO 2:

Objetivos:

2.1 Especificaciones del proyecto

El objetivo principal del proyecto ha sido desarrollar un programa que pudiese recoger datos sobre la actividad diaria de las personas en el hogar. Para poder realizarlo se ha necesitado trabajar sobre unos objetivos secundarios:

- Se ha llevado a cabo una revisión bibliográfica sobre las técnicas existentes para detectar de forma automatizada la calidad de vida de las personas mayores en su hogar.
- Se ha implementado un prototipo de sistema de monitorización mediante Ploggs que sirven para medir el consumo eléctrico, sensores de presencia y sensores de temperatura. Se han elegido estos dispositivos porque son sencillos, no invasivos y de fácil instalación sin requerimiento de obras.
 - Para trabajar con los Ploggs [6, 7, 8], se ha desarrollado una aplicación que recoge la información generada por los Plogg y se almacena en el ordenador.
 - En el caso de los sensores de presencia y temperatura, se han utilizado unos dispositivos multisensores así como una aplicación de recogida de datos desarrollada en Tecnodiscap.
 - Todos los elementos se han comunicado mediante protocolo Zigbee, utilizando el terminal de Telegesis [9].

- Se ha integrado el sistema desarrollado en un hogar para hacer una recogida de datos. De esta manera, se ha intentado conseguir una representación real aunque fuera de forma piloto.
- Una vez obtenidos los datos, se han comparado con registros obtenidos de forma manual para poder comprobar si los datos obtenidos han sido válidos para reconocer las actividades de los usuarios en el hogar.

2.2 Especificaciones técnicas:

Para alcanzar el objetivo de reconocimiento de actividad se han definido los siguientes objetivos parciales:

- Realizar un estudio del estado del arte del reconocimiento de actividades y de la dependencia de forma automatizada.
- Aprender a manejar Matlab [10], como herramienta principal de desarrollo de las aplicaciones.
- Comprender el funcionamiento del protocolo Zigbee [11].
- Saber usar el terminal de Telegesis para el control de las redes de Zigbee.
- Conocer el funcionamiento de los Ploggs.
- Aprender el funcionamiento de los multisensores.
- Desarrollar en Matlab una aplicación que pida y almacene toda la información que generan los Ploggs.
- Diseñar en Matlab una aplicación que procese los datos y muestre las actividades desarrolladas.
- Poner en marcha la recogida de datos de manera continua durante varios días.
- Analizar los datos obtenidos y buscar actividades desarrolladas.

2.3 Distribución temporal del desarrollo del proyecto

A continuación (Figura 1) se muestra la distribución temporal del proyecto con un diagrama de Gantt. Las tareas que se presentan han cumplido los objetivos mencionados anteriormente.

Id.	Nombre de tarea	Comienzo	Fin	Duración	T1	T2		T3			T4	
					mar	abr	may	jun	jul	ago	sep	oct
1	Planificación del proyecto	01/03/2011	12/03/2011	11d	●							
2	Revisión bibliográfica	01/03/2011	27/05/2011	76d	■	■	■	■				
3	Desarrollo del Script de recogida de datos	15/04/2011	05/10/2011	149d		■	■	■	■	■	■	■
4	Diseño de la ubicación de los sensores	10/05/2011	25/06/2011	41d		■	■	■				
5	Desarrollo del Script de tratamiento de datos	05/10/2011	05/11/2011	28d							■	■
6	Recogida de datos	11/10/2011	15/10/2011	5d							●	
7	Análisis de los datos	17/10/2011	07/11/2011	19d							■	■
8	Redacción de la memoria	25/07/2011	16/11/2011	99d				■	■	■	■	■
9	Preparación de la presentación y defensa	18/11/2011	10/12/2011	20d								■

Figura 1. Distribución temporal del proyecto

CAPÍTULO 3:

Estudio del estado actual del arte

La primera parte de mi proyecto ha consistido en la realización de una revisión bibliográfica exhaustiva sobre el tema de estudio.

El objetivo principal ha sido documentarme acerca de la aplicación de tecnologías electrónicas para la obtención de registros de actividades de la vida de las personas que repercuten en la calidad de vida y en su dependencia. Se pretendía conocer qué estudios habían realizado otros autores en este campo.

Para realizar la búsqueda de esta información, visité la página web de Google académico [12] y consulté a través de la biblioteca la base de datos de ISI Web of Knowledge [13]. A pesar de esto, la plataforma de la que he obtenido la mayoría de la información ha sido ieeexplore [14], que además hice un curso en la biblioteca para aprender a utilizarla y buscar correctamente.

A continuación, se ha escrito un breve resumen, centrado en la metodología utilizada, de los artículos más destacados para el reconocimiento de actividad.

En el artículo: “Health smart home for elders - a tool for automatic recognition of activities of daily living” [15].

En este estudio se ha intentado reconocer las actividades de las personas mayores mediante sensores de presencia por infrarrojos (PIR) y contactores magnéticos de puerta en una casa piloto de Grenoble (Francia).

Para el reconocimiento de esas actividades, se comprobó previamente que el sistema reconocía la presencia de la persona en una habitación y que recogía

correctamente las actividades que realizaban. Dependiendo de si el sensor había registrado o no actividad y mediante las secuencias de actividad-inactividad iban deduciendo las actividades diarias de las personas.

Para realizar esta clasificación se utilizó el método de Katz (que recopila las acciones de bañarse, vestirse, ir al wc, moverse, alimentarse), al cual se le añadieron dos actividades más, levantarse y acostarse, que también son fundamentales para determinar la calidad de vida de la persona.

A continuación (Figura 2) se observa como se consiguen diferenciar mediante este método las distintas actividades realizadas por una persona durante un mes.

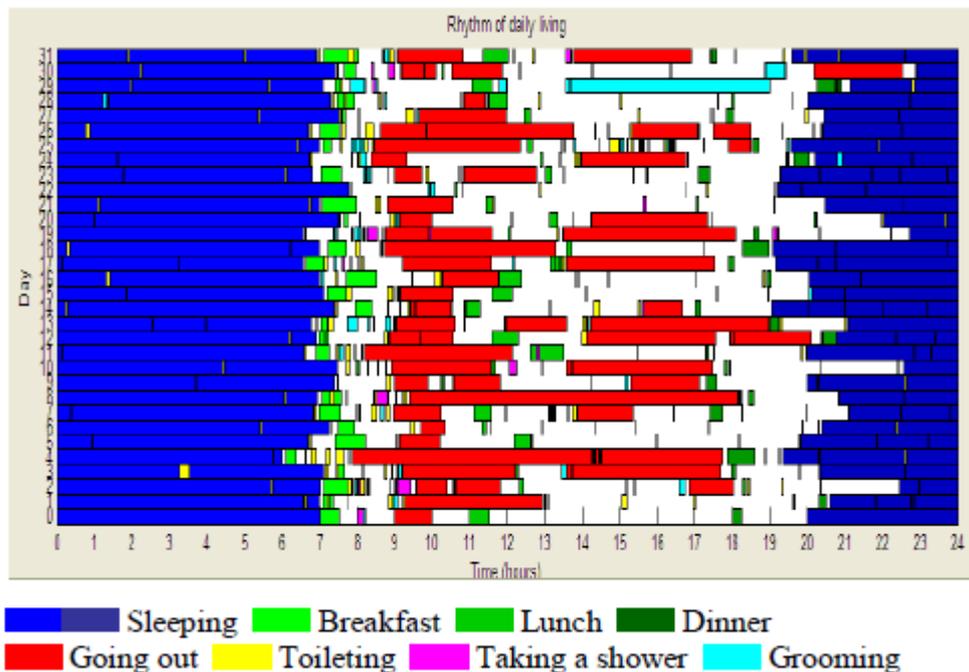


Figura 2. Ritmo de la vida diaria del ocupante durante 31 días

En el artículo: “Smart House Automation System for the Elderly and the Disabled” [16].

En este trabajo se desarrolló un sistema de casa inteligente automatizada para el control de personas mayores y de discapacitados. La finalidad de este sistema ha sido ayudar a vivir de manera independiente, respetando la autonomía de dichas personas.

Para su realización se utilizaron 3 contactores magnéticos, 5 sensores activos de infrarrojos (IR), 6 sensores pasivos IR, 2 sensores de temperatura, 2 pulsadores y 1 sensor de sobreintensidad.

Este experimento se realizó en un piso piloto de Toulouse (Francia), en el cual se colocaron distintos sensores en una habitación de la casa, ya que se consideró que en cada habitación vive una sola persona.

Para realizar el control se entrenó durante un tiempo un sistema de procesamiento basado en redes neuronales con los datos del ocupante de la habitación. Una vez entrenada la red, se comparó los datos obtenidos de los sensores, con los generados por la red neuronal. De esta manera, en caso de discordancia se activaba una alarma.

En el artículo: “Health-status monitoring through analysis of behavioral patterns” [17].

En este trabajo se detectaron patrones de comportamiento mediante sensores de posición instalados en una casa piloto monitorizada de Virginia (Estados Unidos).

Una vez obtenidos los datos, se agruparon y representaron los patrones de comportamiento de cada tipo de evento.

A partir de los eventos se deducían las actividades identificables en este estudio: conducta del sueño, cambio de ropa, ir al cuarto de baño, entrar y salir de casa y comer. Estos eventos constituyen la mayoría de las actividades de la vida diaria, que se utilizan en las evaluaciones funcionales realizadas por los profesionales de la salud.

En este artículo, se consiguió reducir la incertidumbre en la clasificación de las observaciones al evaluar por separado si es un día de trabajo o festivo. Así, se

proporcionó un método para la detección de patrones más fiable que el que tendríamos sin tener en cuenta esa diferenciación.

Con los resultados obtenidos a través de este sistema de análisis de datos, se demostró que se pueden detectar patrones de comportamiento mediante sensores perfectamente adaptables a cualquier hogar a bajo coste.

En el artículo: “Multisensor Fusion for Monitoring Elderly Activities at Home” [18]:

Este experimento se desarrolló en una casa piloto de Francia, dotada de cámaras de video y sensores de distintos tipos (presencia, presión, corriente, caudal y contactores). Este sistema de reconocimiento de actividad es muy preciso y tiene una gran robustez al añadir la detección por imagen. La distribución de estos sensores se muestra a continuación (Figura 3).

Para llevar a cabo este trabajo se requirió la participación de 14 personas, a las cuales se les solicitó que realizaran en el piso piloto una serie de actividades tales como: preparar una comida, tomar una comida, lavar los platos, limpiar la cocina y ver la televisión.

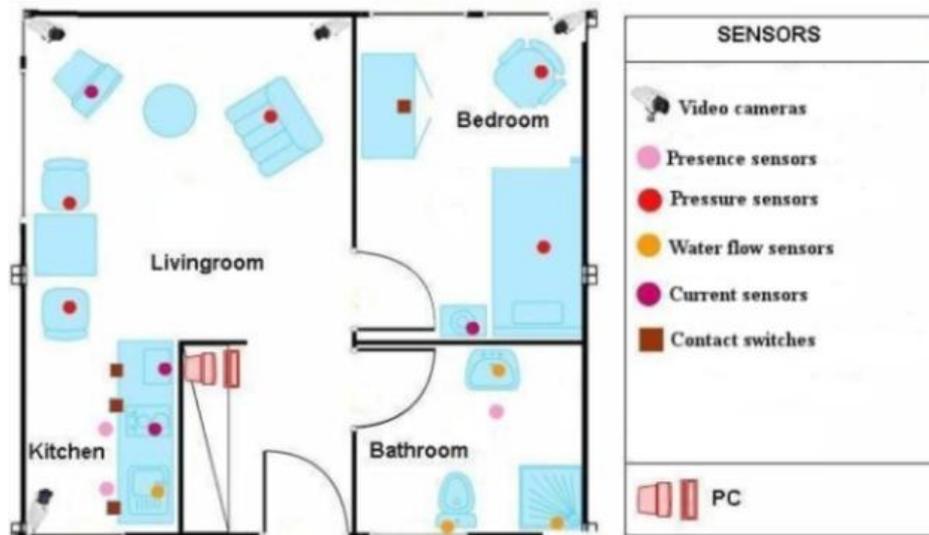


Figura 3. Disposición de los sensores en la casa piloto

Con toda la información recogida por los sensores y las imágenes de video se consiguió representar en tres dimensiones las actividades que realizaron estas personas.

Aunque es un método en el que se detectan muy bien las actividades, tiene el inconveniente de ser invasivo, ya que utiliza cámaras de video. Por tanto, es difícil de implementar en la vida diaria de las personas, debido a la desconfianza que produce estar siendo grabado continuamente.

En el artículo: “SVM-Based Multimodal Classification of Activities of Daily Living in Health Smart Homes Sensors, Algorithms, and First Experimental Results” [19]:

Este trabajo se desarrolló en una casa piloto en Grenoble (Francia). Para ello, se colocaron micrófonos, sensores infrarrojos de presencia, contactores, un sensor de temperatura y humedad, así como una placa que lleva el usuario con acelerómetros de 3 ejes y magnetómetro de 3 ejes. También se utilizaron cámaras de video pero sólo para la supervisión de la máquina de aprendizaje de los algoritmos. La distribución de los sensores se observa a continuación (Figura 4).

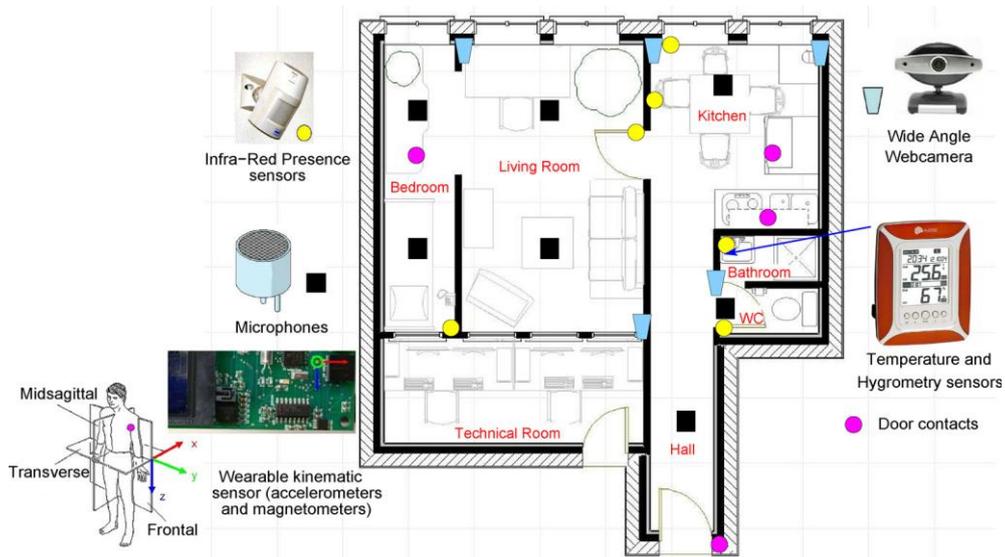


Figura 4. Colocación de los sensores en la casa

En este estudio se buscó detectar una pérdida de autonomía tan pronto como fuese posible, por este motivo, se detectaron siete actividades: higiene, usar el baño, comer, descansar, dormir, comunicación, y vestirse / desvestirse. Tras el tratamiento de los datos se obtuvieron unos resultados especialmente buenos en el reconocimiento de las actividades planteadas (Tabla 1).

		Classification Results						
		Sleeping	Resting	Dress/undress	Eating	Toilet use	Hygiene	Communication
Activity	Sleeping	93.9%	6.1%	0%	0%	0%	0%	0%
	Resting	13.8%	78.1%	1.3%	1.3%	4.2%	1.3%	0%
	Dressing/undressing	6.25%	12.5%	75%	0%	0%	0%	6.25%
	Eating	0%	0%	2.2%	97.8%	0%	0%	0%
	Toilet use	0%	0%	0%	6.25%	93.75%	0%	0%
	Hygiene	7.1%	0%	0%	7.1%	21.5%	64.3%	0%
	Communication	0%	5.3%	5.3%	5.9%	0%	0%	89.5%

Tabla 1. Resultados del reconocimiento de actividades

En este caso, aunque las cámaras no fueron utilizadas para la detección de actividad, al haber micrófonos, se considera un método invasivo. A pesar de esto, es muy interesante porque dio unos buenos resultados con el añadido de que puede detectar peligros o posibles caídas con los micrófonos puesto que tiene memorizadas palabras claves.

El inconveniente que tiene es, que se necesitan bastantes recursos y dinero ya que para procesar la información se requiere cuatro ordenadores, además de disponer de un piso con todos los sensores cableados.

En el artículo: “An Activity Monitoring System for Real Elderly at Home: Validation Study” [20].

El experimento que describe este artículo se desarrolló en un apartamento piloto de Sophia Antipolis (Francia).

Se realizó con nueve personas voluntarias de edades comprendidas entre 64 y 85 años, siendo totalmente vigilados cada uno durante cuatro horas.

En el mobiliario de la casa se incorporó cámaras de video y distintos tipos de sensores ambientales que registraron los datos de temperatura, humedad, agua, tensión

eléctrica, luz, de presión (en camas, sillas y sillones) y de apertura/cierre de puertas, ventanas y cajones. De esta manera, se consiguió un control total de todas las actividades que realizaron los individuos obteniendo una gran precisión en el reconocimiento de dichas actividades. Además se logró una diferenciación de resultados, detectando falsos positivos (en el reconocimiento de “lavar los platos” se detectó esta actividad cuando alguien abría el grifo del agua para lavarse las manos) y falsos negativos (cuando una persona lavaba una taza que lleva menos tiempo que el hecho de lavar los platos) (Tabla 2). Un inconveniente de esta aplicación es que se considera un método invasivo de la intimidad de los usuarios.

Multimodal activity	GT	TP	FN	FP	P	S
<i>Use fridge</i>	65	54	11	9	86%	83%
<i>Use stove</i>	177	165	11	15	92%	94%
<i>Sitting on chair</i>	66	54	12	15	78%	82%
<i>Sitting on armchair</i>	56	49	8	12	80%	86%
<i>Prepare lunch</i>	5	4	1	3	57%	80%
<i>Wash dishes</i>	16	13	3	7	65%	81%

Tabla 2. Reconocimiento de actividades

Siendo:

GT- Verdaderos del programa.

TP- Verdaderos positivos.

FN- Falsos negativos.

FP- Falsos positivos.

P- Precisión ($P=TP/(TP+FP)$).

S- Sensibilidad ($S=TP/(TP+FN)$).

Como futuros trabajos se ha planteado poner a prueba este método durante más tiempo así como su implantación en hospitales.

En el artículo: “An Algorithm for the Automatic Detection of Health Conditions [21]”.

En este artículo se desarrolló un algoritmo para el diagnóstico de la salud de las personas mayores que viven solas mediante una aplicación económica que trabaja con datos sencillos.

Para el diseño de este algoritmo sólo se tuvo en cuenta los datos del consumo de la televisión. Esto es así, porque se supuso que una persona suele ver la televisión con una cierta rutina todos los días, ya que ve los mismos programas cuando está bien de salud. Por el contrario, en caso de enfermedad se cambia esta rutina.

El experimento duró 7 meses y se realizó con una sola persona en una casa de Japón. Los resultados obtenidos se validaron mediante un cuestionario que se le pasó a la persona.

La prueba consistió en recoger los datos de consumo de la televisión a lo largo del día. Con esos datos se construyó una imagen monocromática. Para ello, se dividió las 24 horas del día en periodos de 15 minutos y con esos datos se logró crear una imagen de 30 x 96 píxeles. Esta imagen representa un mes. A partir de estas imágenes, se consiguió diagnosticar el estado de salud del sujeto mediante la entropía de las imágenes obtenidas.

A continuación se muestran las imágenes de los 7 meses que duró el ensayo en las cuales se puede observar como el último mes es muy distinto a los anteriores (Figura 5). Ese mes coincidió con que el usuario estaba enfermo, según los resultados de la encuesta.

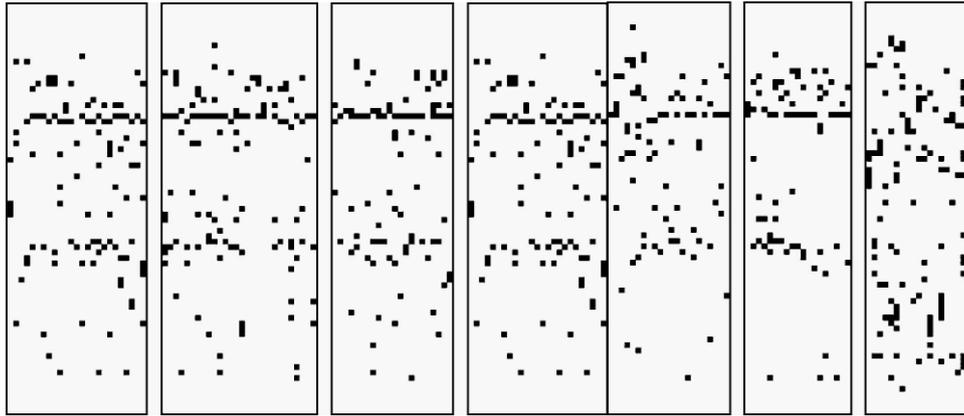


Figura 5. Imágenes del tiempo que vio la televisión

Este algoritmo también puede ser aplicado con otros patrones de la vida diaria que tengan unas horas de comienzo más o menos fijas, como puede ser la comida ya que las personas mayores suelen realizar las comidas casi siempre a la misma hora.

La ventaja de este método es que podría utilizarse en la evaluación de un diagnóstico para la prevención de enfermedades, que posteriormente, tendría que confirmar el personal sanitario.

En el artículo: “Long-term remote behavioral monitoring of the elderly using sensors installed in domestic houses” [22].

En este ensayo se controló el comportamiento diario de dos señoras mayores que vivían solas en Mizusama (Japón) durante un periodo superior a un año.

Para eso, se instalaron en sus propias casas varios sensores: infrarrojos (controlar el movimiento de la persona), magnéticos (detectar apertura/cierre de puertas), vatímetros integrados en las tomas de corriente (detectar el uso de electrodomésticos), detectores de llama (detectar el uso de los fogones de la cocina) y un sensor de dióxido de carbono (controlar la presencia de la persona en la habitación por el CO₂ exhalado). Estos sensores se conectaron a un ordenador mediante una red cableada.

Los datos se transfirieron de forma automática a Internet todos los días.

A partir de los datos recogidos se recopiló información sobre patrones de conducta diaria, tales como la duración del sueño, la entrada y salida de casa, el uso de

los fogones de la cocina, la apertura/cierre de puertas o el tiempo dedicado a ver la televisión.

Con esta vigilancia se ha conseguido contribuir al mantenimiento de la salud de las personas mayores.

CAPÍTULO 4:

Desarrollo del proyecto

4.1 Diseño de aplicaciones:

4.1.1. Introducción

Para conseguir un sistema que permitiera monitorizar la actividad de una persona, se realizó un estudio de las herramientas que hay disponibles para tal fin. Por eso, se requirió aprender a manejar programas como Matlab, Telegesis Terminal, así como las comunicaciones Zigbee, los Ploggs o los multisensores.

Después, se realizó un análisis para determinar la ubicación de los distintos sensores en el hogar en función de las actividades que se querían determinar. También se desarrolló un Script que generaba un archivo con los datos de consumo de los Ploggs.

Cuando ya se implementaron las herramientas de recogida de datos, se instalaron los sensores en las distintas habitaciones del hogar y procedió a la recogida de datos durante un periodo de tiempo.

Por último, se necesitó desarrollar un Script que permitiese procesar los datos obtenidos. Mediante este programa y la información recogida de los sensores durante las pruebas realizadas, se pudo traducir a actividades diarias y representarlas gráficamente.

4.1.2. Diagrama de bloques

El diseño del desarrollo del prototipo que queríamos realizar se presenta a continuación (Figura 6). En ella, se observa un diagrama físico que da una idea general del sistema completo y de sus bloques funcionales.

En primer lugar, se encuentra el Hardware, constituido por los Ploggs y los sensores, tanto de presencia, como de temperatura. Previamente a la instalación de estos en el hogar, se diseñó su ubicación idónea para detectar la mayor cantidad de actividades posibles. Dichos sensores se conectaron con el ordenador mediante comunicación Zigbee.

A partir de aquí, estaría el desarrollo del Software, que se realizó en el ordenador. Se debían programar unos Scripts para la recogida de datos. Una vez almacenados estos datos en ficheros de texto, se realizó su posterior tratamiento mediante otro Script que es el que consiguió traducir los datos en actividades. De esta manera, se cumplió nuestro objetivo principal.

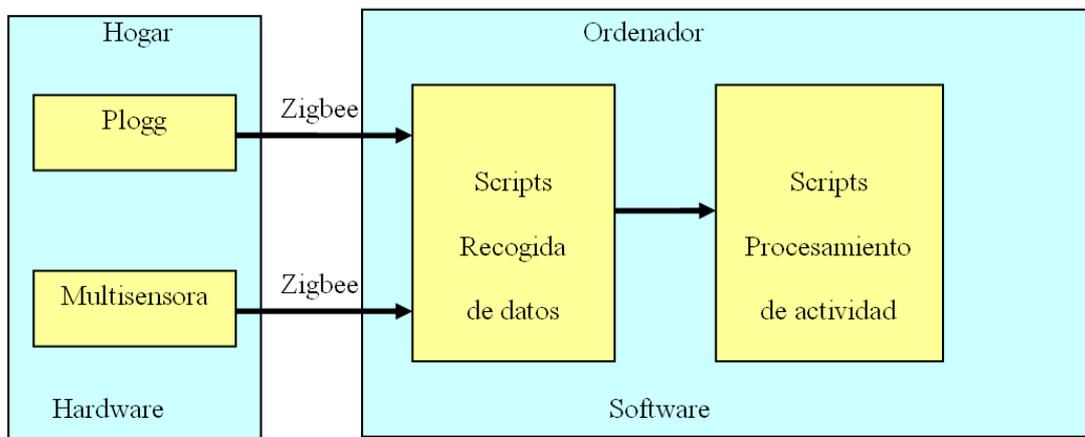


Figura 6. Diagrama del sistema

Para llevar a cabo este diseño se siguió el siguiente procedimiento (Figura 7). En primer lugar, se trabajó desarrollando el script de recogida de datos. Una vez hecho, se probaba y en caso de fallo se volvía a la fase del desarrollo de ese script. De forma paralela, se trabajaba en el diseño de la correcta ubicación de los sensores. De igual

modo, se pensaba si la información era útil o no. Si no lo era, se cambiaban los sensores de ubicación.

Una vez desarrollados estos dos procesos se procedió a la recogida de datos. Durante la recogida y una vez terminada esta, se creó el Script de tratamiento de datos. Se fue probando hasta que se consiguió que diera datos de actividad. Con este script ya desarrollado, se pasó a la fase de analizar los datos y detectar actividad.

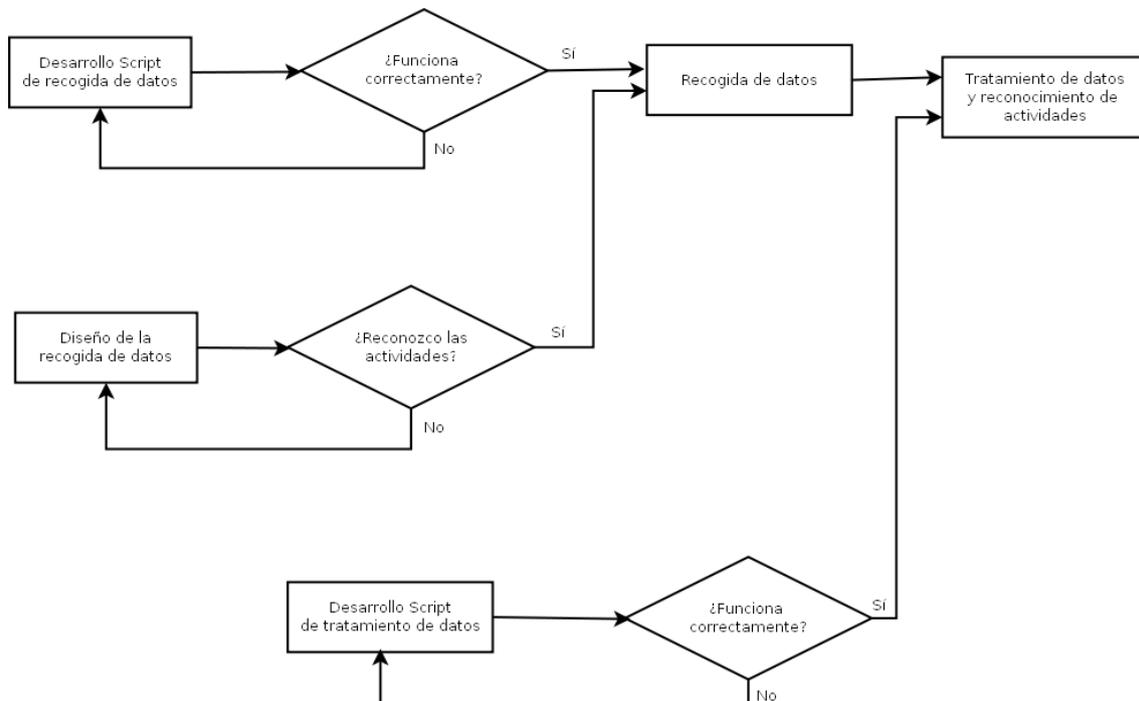


Figura 7. Diagrama del proceso de desarrollo de la herramienta

4.1.3. Herramientas globales

A continuación se han descrito las principales herramientas informáticas que he tenido que aprender para realizar este proyecto. Por un lado, Matlab es un programa con el que se ha desarrollado todo el software y por otro, el Telegesis Terminal que ha sido empleado para las comunicaciones Zigbee. A parte de éstos, ha sido necesario instalar unos drivers en el ordenador.

4.1.3.1. Matlab:

Matlab [23, 24, 25, 26] es un lenguaje de alto nivel para computación técnica. Éste integra computación, visualización y programación, en un entorno fácil de usar donde los problemas y las soluciones son expresados con notación matemática. Los usos de Matlab son:

- Matemática y Computación.
- Desarrollo de algoritmos.
- Modelamiento, simulación y prototipado.
- Análisis de datos, exploración y visualización.
- Graficas científicas e ingenieriles.
- Desarrollo de aplicaciones, incluyendo construcción de interfaces gráficas de usuario.

Matlab es un sistema interactivo cuyo elemento básico de almacenamiento de información es la matriz. Tiene una característica fundamental y es que no necesita dimensionamiento. Esto, le permite resolver varios problemas de computación técnica (especialmente aquellos que tienen formulaciones matriciales y vectoriales) en una fracción de tiempo similar al que se gastaría cuando se escribe un programa en un lenguaje no interactivo como C.

El nombre Matlab simboliza Matrix Laboratory o Laboratorio de Matrices. Matlab fue originalmente escrito para proveer fácil acceso el software de matrices desarrollado por los proyectos Linpack y Eispack.

Matlab es una aplicación instructiva estándar para cursos avanzados e introductorios en matemáticas, ingeniería y ciencia. Además es una de las principales herramientas utilizadas en la investigación científica. En la industria Matlab es la herramienta escogida para investigación de alta productividad, desarrollo y análisis.

Matlab presenta una familia de soluciones a aplicaciones específicas de acoplamiento rápido llamadas ToolBoxes. Los ToolBoxes son colecciones muy

comprensibles de funciones Matlab o archivos de Matlab (M-files) que extienden el entorno de Matlab para resolver clases particulares de problemas. Algunas áreas en las cuales existen ToolBoxes son:

- Procesamiento de señales.
- Sistemas de control.
- Redes neuronales.
- Lógica difusa.
- Wavelets.
- Simulación.

El sistema Matlab consta de cinco partes principales:

Entorno de desarrollo:

Es el conjunto de herramientas y módulos que ayudan a usar las funciones y archivos de Matlab. Muchas de esas herramientas son interfaces gráficas de usuario (Figura 8). Esto incluye: el escritorio de Matlab, la ventana de comandos, el historial de comandos, un editor y un depurador, navegadores para revisión de la ayuda, el espacio de trabajo o Workspace y los archivos.

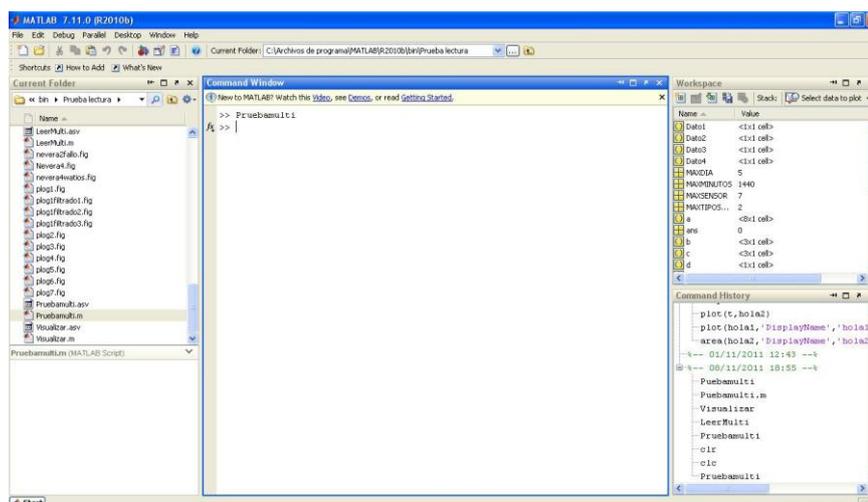


Figura 8. Interface de Matlab

La librería de funciones matemáticas:

Constituida por una gran colección de algoritmos computacionales que van desde funciones elementales como la suma, la función seno y coseno, y la aritmética de números complejos hasta funciones mucho más sofisticadas como inversas de matrices, autovalores de matrices y transformadas radiadas de Fourier.

El lenguaje Matlab:

Es un lenguaje de alto nivel para matrices con sentencias para control de flujo, creación de funciones y estructuras de datos, funciones de entrada/salida y algunas características de programación orientada a objetos. Este lenguaje permite, tanto la programación a pequeña escala para la creación rápida de programas, como la realización de aplicaciones complejas.

Gráficas:

Matlab cuenta con módulos extensivos para la visualización de vectores y matrices en forma de gráficas, así como para realizar comentarios e impresión de estas gráficas. Matlab incluye funciones de alto nivel para la visualización de datos en dos y tres dimensiones, procesamiento de imágenes y animación. También incluye funciones de bajo nivel que permiten personalizar completamente la apariencia de los gráficos así como construir interfaces graficas de usuario para las aplicaciones.

Interfaces Externas:

Las interfaces externas son un conjunto de librerías que permiten la programación en lenguaje C y Fortran de programas que interactúen con Matlab. Estas librerías facilitan realizar llamadas de rutinas desde Matlab.

Ayuda para Matlab

Matlab dispone de una documentación extensiva, tanto en formato impreso como en línea, para ayudar a los usuarios a aprender todas sus características. La ayuda

que incorpora Matlab provee información orientada a tareas e información de referencia acerca de todas las características de Matlab.

Para ver la documentación online de Matlab se selecciona la opción “Ayuda de Matlab” del menú Ayuda en la barra de menús de Matlab. En esta ayuda se puede obtener información de todas las funciones que reconoce Matlab y como utilizarlas.

También es muy útil la ayuda que se puede encontrar en foros de uso de Internet, así como los videos y demostraciones de la web de Matworks [27, 28, 29].

4.1.3.2. Telegesis Terminal:

Telegesis Terminal es una aplicación basada en un interfaz gráfico que permite al usuario comunicarse con los módulos inalámbricos de Telegesis, tanto mediante el envío de comandos a los dispositivos como recibiendo la información que estos generan (Figura 9).

El usuario puede personalizar la interfaz gráfica del Telegesis Terminal para adaptarlo a sus necesidades. Los comandos utilizados con más frecuencia se pueden agregar en forma de botones, así como eliminarlos cuando ya no sean útiles. También se puede crear un registro de la actividad que se genera para verlo más tarde.

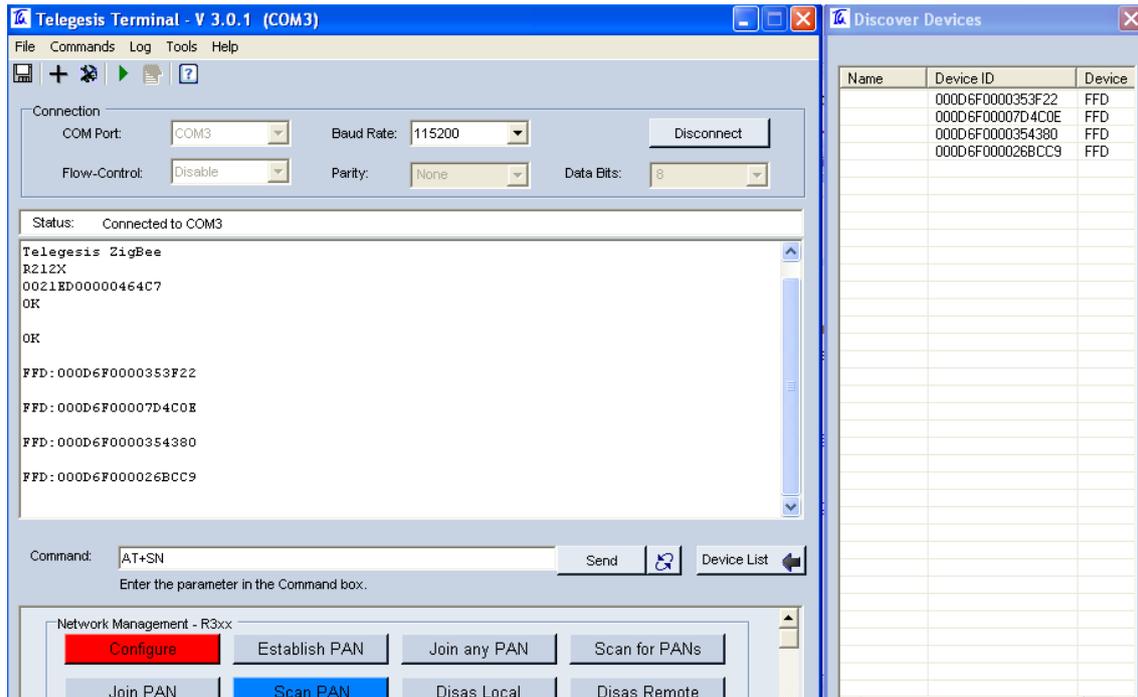


Figura 9. Terminal de Telegesis

Se ha elegido esta herramienta para configurar y administrar las redes Zigbee que se han empleado en el proyecto, tanto la de los Ploggs como la de los multisensores. Se han tenido que hacer dos redes porque llevan distintas versiones de Zigbee, los Plogg llevan la R212 y los multisensores la R302. Además, estas versiones son incompatibles entre sí.

La aplicación tiene las siguientes características:

- Configurar comandos: agregar, actualizar y eliminar botones de comando.
- Registrar configuraciones: guarda las actividades registradas en el archivo de registro.
- Guardar datos: guarda los comandos generados por los usuarios en relación con los botones en un archivo.
- Enviar archivo: enviar n archivos a un dispositivo de hardware.

El Flujo de la aplicación es el siguiente:

- Agregar, actualizar y eliminar comandos.
- Conectarse a los equipos inalámbricos.
- Guardar el estado actual de la ventana.
- Guardar el archivo de registro.
- Ver el registro en el Bloc de notas.
- Transferir archivos a un dispositivo hardware.

Para la comunicación con los módulos Zigbee de Telegesis se deben enviar las instrucciones a ejecutar con las acciones requeridas. El dispositivo de hardware conectado ejecuta ese comando y devuelve el resultado. Los comandos disponibles están predefinidos en la aplicación. Sin embargo, se pueden editar otros comandos. También se puede enviar un comando parametrizado a un módulo deseado.

Telegesis Terminal permite cambiar entre los distintos puertos COM detectados, para poder configurar distintas redes desde un mismo ordenador. Además, hay una lista con la identificación de los dispositivos que están disponibles en la red y desde esta es fácil parametrizarlos.

A continuación, se muestran las principales instrucciones (Tabla 3). Para profundizar más, en los anexos se incluyen los comandos de la versión TG-ETRX-R212-AT.

Command Overview	
ATI	Display product identification information
ATZ	Software reset
AT&F	Restore factory settings
AT+BLOAD	Enter the bootloader menu
AT+CLONE	Clone the local node's firmware to a remote node
AT+RECOVER	Recover from a failed clone attempt (ETRX2 only)
AT+PASSTHROUGH	Enter pass-through bootloading mode (ETRX1 only)
ATS	S-Register access

ATSALL	Write all remote S-Registers
AT+TOKDUMP	Display all local S-registers
ATSREM	Remote S-register access
AT+ESCAN	Scan the energy of all channels
AT+EN	Establish PAN
AT+JN	Join next best network
AT+PANSCAN	Scan for active PANs
AT+JPAN	Join specific PAN
AT+DASSL	Disassociate local device from PAN
AT+DASSR	Disassociate remote device from PAN
AT+NTABLE	Show the neighbour table
AT+N	Display network parameters
AT+CTABLE	Display list of local children
AT+PARENT	Display Parent's ID
AT+POLL	Poll Parent for data
AT+SN	Scan network for other nodes
AT+REMSN	Scan for remote device's direct neighbours
AT+LINKCHECK	Check link parameters with a neighbour
AT+PING	Indicate presence in the network
AT+BCAST	Transmit a broadcast
AT+BCASTB	Transmit a broadcast of binary data
AT+UCAST	Transmit a unicast
AT+UCASTB	Transmit a unicast of binary data
AT+SCAST	Transmit data to the Sink
AT+SCASTB	Transmit binary data to the sink
AT+SSINK	Search for a sink
AT+SINK	Display the local Node's sink
AT+OPCHAN	Opens a channel to a remote node
+++	Close channel
AT+OPLCHAN	Opens a limited channel to a remote node
AT+ACKCHAN	Accept channel
AT+RDATAB	Send binary raw data
AT+IDENT	Play a tune on remote devboard

Tabla 3. Comandos Zigbee

4.1.3.3. Instalación drivers:

Entre otros se han tenido que instalar todos los drivers para el perfecto funcionamiento del Telegesis Terminal así como todos los Microsoft .NET Framework requeridos. Para la instalación de los dispositivos controladores de la red Zigbee se requería tener instalada o instalar la versión de “Microsoft .NET Framework Version 1.1”. En la dirección de Internet de Microsoft estaba disponible la última versión, pero no se encontraba la que requería la aplicación. Al final la conseguí encontrar.

4.2 Hardware:

En las páginas siguientes, se han descrito los sensores empleados, que han sido, por un lado el Plogg que mide el consumo de la red eléctrica de los aparatos conectados a él y por otro, los multisensores que constan de entre otros, un sensor de posición y otro de temperatura. Todos ellos se comunicaron con el ordenador mediante una red inalámbrica usando el protocolo Zigbee.

Para llevar a cabo la realización del proyecto, se precisó diseñar una buena colocación de los sensores en el hogar que permitiera recoger la información para poder deducir las actividades diarias de la persona.

4.2.1. Plogg:

El Plogg es un dispositivo simple, de bajo coste y fácil instalación que informa y controla en tiempo real el consumo energético de cualquier tipo de dispositivo o sistema eléctrico (Figura 10).

Las principales características técnicas son:

- Medir una sola fase de la distribución eléctrica.
- Disponer de comunicación inalámbrica Zigbee para comunicarse con el ordenador y dar la información que este le pida.
- Usar un microprocesador para realizar las medidas y controlar las comunicaciones con el módulo de radio inalámbrico que es un Teridian 6511.

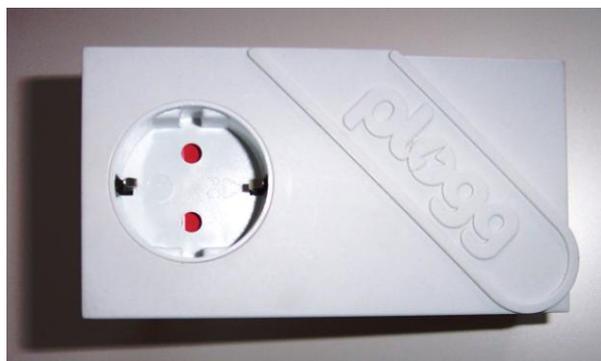


Figura 10. Plogg

El protocolo que se utiliza para transmitir los datos al ordenador a través del Zigbee puede ser o ASCII o binario. Y los comandos que recibe el Plogg con las instrucciones son ASCII.

Este dispositivo, mide la potencia activa instantánea, la energía activa consumida, la potencia reactiva, la tensión, la intensidad, la potencia reactiva consumida, la energía activa generada, la frecuencia, el ángulo de fase, la potencia reactiva generada y el tiempo de encendido del aparato.

En la comunicación Zigbee la instrucción se manda mediante un AT+UCAST:<La dirección MAC del dispositivo=la instrucción> como por ejemplo:
AT+UCAST:000D6F000026C7ED=YV.

La forma en que se recibe el dato es <una etiqueta><longitud del paquete de datos binarios><El paquete de datos binarios>. La etiqueta es un número que se incrementa para cada UCAST enviado, lo que permite comprobar que no se han perdido datos y mantener todas las respuestas en el orden correcto. La longitud del paquete de datos binarios también es un número de valor entre 1 y 64 en decimal. El paquete de datos binarios varía en función de la orden que se ha pedido.

Los principales comandos son:

YV- Muestra la información instantánea de consumos.

YS- Precios de la tarifas.

YT- Tiempos de las tarifas.

YI- Intervalo del registro de datos.

YC- Muestra el coste acumulado del consumo.

YR- Resetea los valores almacenados de consumo.

YM- Muestra los valores máximos.

YD- Muestra los valores almacenados.

YX- Elimina los valores almacenados.

YL- Selecciona los valores del registro.

YO- Enciende o apaga la temporización.

YE- Muestra la información de temporización.

YN- Da nombre al dispositivo.

ZN- Configuración de identificación de red.

ZR- Configuración como enrutador.

ZC- Configuración como coordinador.

ZH- Configuración del número del canal.

Para resetear el Plogg y ponerlo a los valores de fábrica hay que conectarlo 5 segundos y desconectarlo otros cinco segundos repitiendo este proceso tres veces, y al cabo de unos treinta segundos se escuchan unos relés.

En los anexos se ha añadido la guía con los datos del protocolo binario.

4.2.2. Multisensores:

El multisensor es un dispositivo constituido por una serie de sensores. Este sistema desarrollado por Tecnodiscap, manda la información recogida a través de un módulo de comunicación Zigbee. Entre todos los sensores que tiene, para este proyecto se han usado dos, el de presencia y el de temperatura (Figura 11).



Figura 11. Multisensor

El sensor de presencia es un Panasonic MP Motion Sensor [30]. Detecta leves movimientos a una distancia de hasta dos metros y tiene un consumo muy bajo de corriente 46 mA. Además, lleva un encapsulado metálico TO-5 para aumentar su inmunidad al ruido provocado por las ondas electromagnéticas ambientales.

El sensor de temperatura es un LM75 [31] con un paso de temperatura de 0.5°C. Además, está preparado para una comunicación serie síncrona I2C. La resolución de éste sensor es de 9 bits. Tiene una interface de dos hilos y dispone de un perro guardián térmico.

4.2.3. Comunicación Zigbee:

4.2.3.1. Introducción

Zigbee [32, 33, 34, 35] es un estándar de comunicaciones inalámbricas diseñado por la Zigbee Alliance. Es un conjunto estandarizado de soluciones que pueden ser implementadas por cualquier fabricante. Zigbee está basado en el estándar IEEE 802.15.4 de redes inalámbricas de área personal (Wireless Personal Area Network, WPAN). Tiene como objetivo las aplicaciones que requieren comunicaciones seguras con baja tasa de envío de datos y maximización de la vida útil de sus baterías. Es un sistema ideal para redes domóticas, específicamente diseñado para reemplazar la proliferación de sensores/actuadores individuales. La razón de ello son diversas características que lo diferencian de otras tecnologías:

- Su bajo consumo
- Su topología de red en malla
- Su fácil integración (se pueden fabricar nodos con muy poca electrónica).

4.2.3.2. Estándar IEEE 802.15.4

IEEE 802.15.4 es un estándar que define el nivel físico y el control de acceso al medio (Figura 12) de redes inalámbricas de área personal con bajas tasas de transmisión de datos (Low-Wireless Personal Area Network, LR-WPAN).

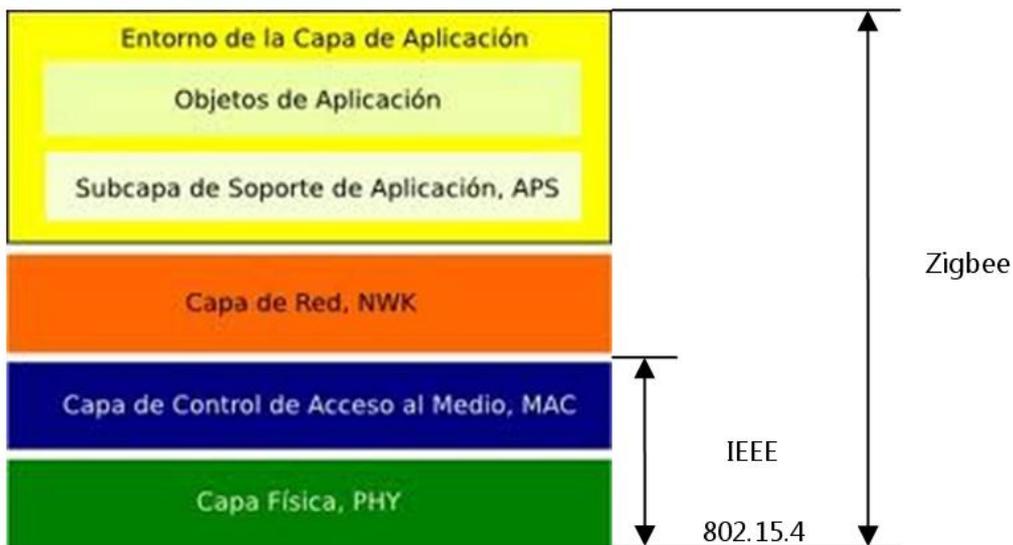


Figura 12. Estructura funcional de Zigbee sobre IEEE 802.15.4

El control de acceso al canal lo realiza mediante CSMA/CA (Carrier Sense Multiple Access, Collision Avoidance) y opcionalmente GTS (guaranteed time slot). La primera técnica consiste en que cada nodo, antes de transmitir, escucha el canal. Si la potencia que detecta está por encima de un cierto nivel, asume que otro nodo está transmitiendo y por lo tanto espera para realizar su emisión. La segunda técnica, implementa un sistema por el cual el coordinador asigna unos slots de tiempo a cada nodo en los que tienen asegurado el acceso limpio al canal.

Utiliza asimismo la confirmación explícita de mensajes recibidos por cada nodo, ACK (message acknowledgement), dado que al estar pensado para bajas tasas de transmisión no hay problema de sobrecarga del medio con dichos ACKs.

Hay tres bandas ISM en las cuales puede trabajar:

- 2,4 GHz: 16 canales, 250 kbps (a nivel mundial).
- 868,3 MHz: 1 canal, 20 kbps (banda libre en Europa).
- 902 – 928 MHz: 10 canales, 40 kbps (banda libre en EE.UU.).

4.2.3.3. Tipos de dispositivos

Se definen tres tipos distintos de dispositivo Zigbee según su papel en la red:

- **Coordinador Zigbee (Zigbee Coordinator, ZC):** El tipo de dispositivo más completo. Debe existir solamente uno por red. Sus funciones son las de encargarse de controlar la red y los caminos que deben seguir los dispositivos para conectarse entre ellos. Requiere memoria y capacidad de computación.
- **Router Zigbee (Zigbee Router, ZR):** Interconecta dispositivos separados en la topología de la red, además de ofrecer un nivel de aplicación para la ejecución de código de usuario. No puede estar dormido.
- **Dispositivo final (Zigbee End Device, ZED):** Posee la funcionalidad necesaria para comunicarse con su nodo padre (el coordinador o un router) pero no puede transmitir información destinada a otros dispositivos. De esta forma, este tipo de nodo puede estar dormido la mayor parte del tiempo, aumentando la vida media de sus baterías. Un ZED tiene requerimientos mínimos de memoria y es por tanto, significativamente más barato.

Basándose en su funcionalidad desde el punto de vista de IEEE 802.15.4, puede plantearse una segunda clasificación:

- **Dispositivo de funcionalidad completa (FFD):** También conocido como nodo activo. Es capaz de recibir mensajes en formato 802.15.4. Gracias a la memoria adicional y a la capacidad de computar, puede funcionar como Coordinador o Router Zigbee o puede ser usado en dispositivos de red que actúen de interfaces con los usuarios.
- **Dispositivo de funcionalidad reducida (RFD):** También conocido como nodo pasivo. Tiene capacidad y funcionalidad limitadas con el objetivo de conseguir un bajo coste y una gran simplicidad.

Un nodo Zigbee (tanto activo como pasivo) reduce su consumo gracias a que puede permanecer dormido la mayor parte del tiempo (incluso muchos días seguidos). Cuando se requiere su uso, el nodo ZigBee es capaz de despertar en un tiempo ínfimo, para volverse a dormir cuando deje de ser requerido.

4.2.3.4. Topologías y funcionamiento de la red

Zigbee permite tres topologías de red (Figura 13):

- Topología en estrella: uno de los dispositivos asume el rol de coordinador de red y es responsable de inicializar y mantener los dispositivos en la red. Todos los demás dispositivos Zigbee, conocidos con el nombre de dispositivos finales, hablan directamente con el coordinador.
- Topología en árbol: es un caso especial de topología de conexión punto a punto, en la cual cualquier dispositivo puede actuar como coordinador y proveer servicios de sincronización hacia otros dispositivos o conectarse como un nodo único al final de la red.
- Topología de malla: el coordinador Zigbee es responsable de inicializar la red y de elegir los parámetros de la red, pero ésta puede ser ampliada a través del uso de routers Zigbee. El algoritmo de encaminamiento utiliza un protocolo de pregunta-respuesta (request-response) para eliminar las rutas que no sean óptimas. La red final puede tener hasta 254 nodos. Utilizando el direccionamiento local se puede configurar una red de más de 65000 nodos.

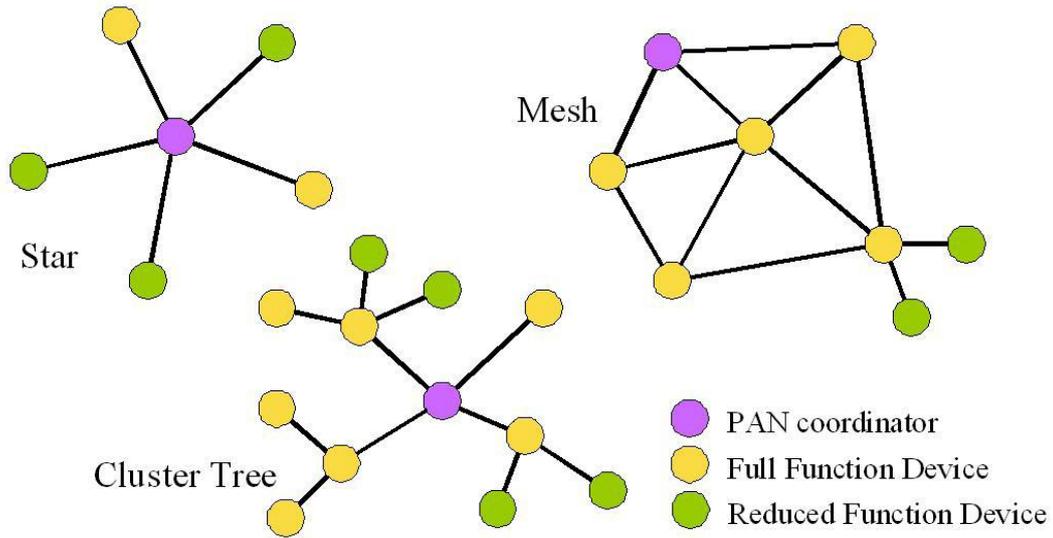


Figura 13. Topologías de red

4.2.3.5. Comunicaciones

Para poder intercambiar mensajes entre los nodos necesitamos, de alguna manera, identificar cada dispositivo dentro de la red. La capa MAC nos ofrece un identificador único de 64 bits o EUI64 (Extended Unique Identifier), que se asigna al chip durante su fabricación y la capa de red otro de 16 bits o NodeID (Node Identifier), quedando la dirección 0x0000 reservada para el coordinador. El coordinador es el responsable de asignar las direcciones del nivel de red según el número máximo de hijos por nodo. Los hijos reciben las direcciones siguientes a la de su padre (Figura 14). Con estos identificadores y gracias a las técnicas de encaminamiento o enrutado utilizadas por los routers, cualquier nodo de la red puede comunicarse con cualquier otro nodo.

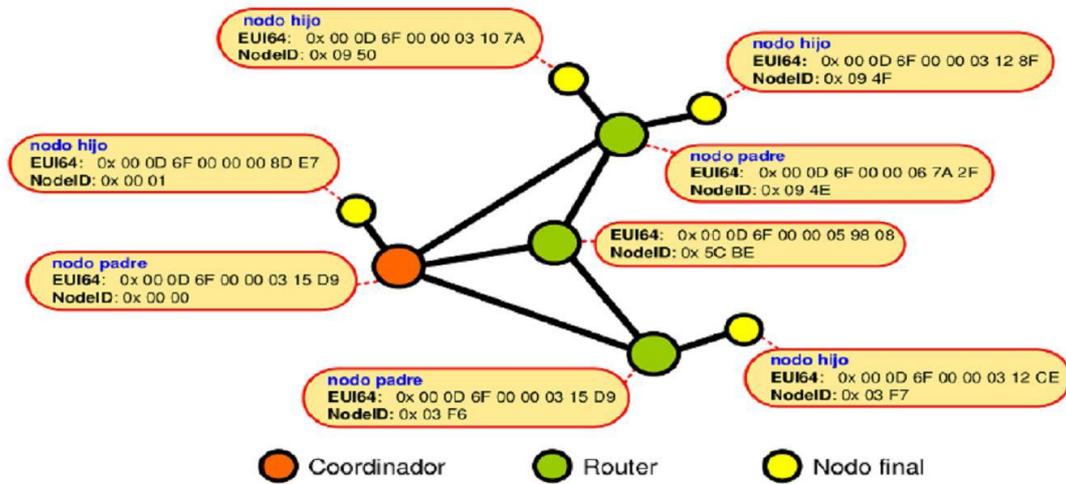


Figura 14. Reparto de direcciones en una red

Los routers reenvían aquellos mensajes que no van dirigidos a ellos. Deben conservar tablas con las rutas descubiertas y deben iniciar o participar en el descubrimiento de nuevas o mejores rutas. También han de ser capaces de detectar y corregir errores. Las tablas contienen información sobre el coste de cada ruta. El coste determina cuál es la mejor en un momento dado. La función que elige el coste de una ruta suele basarse en la latencia del recorrido de los mensajes y en el nivel de potencia detectado en la recepción de los mismos.

Si un mensaje llega a un dispositivo que no es un router, comprueba la dirección de destino y sólo lo reenvía, si va dirigido a alguno de sus hijos. Si es para él, lo pasa a la capa superior. En otro caso se descarta. El nodo padre guarda durante un tiempo, los mensajes dirigidos a sus hijos sleepy o móviles mientras duermen, ya que en ese momento no pueden enviar ni recibir mensajes. Antes de que expire el tiempo el hijo debe despertarse y solicitar el envío de los mensajes pendientes, si los hubiese. En caso contrario, el padre descarta dichos mensajes. Los nodos padre registran la información relativa a sus hijos en su tabla de hijos. Periódicamente el hijo debe enviar un mensaje a su padre para indicar que continúa conectado a él. De no hacerlo, el padre entiende que su hijo ha dejado la red o que le ha abandonado y, en consecuencia, lo borra de su tabla de hijos.

4.2.3.6. Mensajes de datos

Se distinguen cuatro tipos de mensajes para el envío de datos. La principal diferencia entre ellos está en la capa que los genera.

La subcapa de soporte de aplicación (APS) utiliza la dirección local de 16 bits para enviar mensajes a un nodo (mensajes unicast) o a un grupo dentro de un radio de alcance determinado (mensajes broadcast). A su vez, la capa de red permite enviar también mensajes a un único nodo (mensajes datagrama) o a varios nodos (mensajes multicast) que comparten el mismo identificador de grupo utilizando para ambos casos direcciones de 64 bits. Cada nodo guarda la dirección de los nodos con los que se comunica en su tabla de bindings o binding table. La tabla de bindings guarda la dirección EUI64 de un nodo (unicast binding) o un identificador de grupo (multicast binding).

Los nodos finales no pueden recibir ni enviar mensajes broadcast ni multicast. Los mensajes datagrama son los únicos mensajes que reciben confirmación (ACK, Acknowledgement) de haber sido recibidos correctamente en destino.

4.2.3.7. Seguridad

Zigbee permite establecer una clave de seguridad de 128 bits para controlar qué nodos se unen a la red y para encriptar todas las comunicaciones. Un nodo podrá unirse a la red y entender los mensajes sólo si conoce la clave de seguridad.

Utiliza el estándar de encriptación AES-128 (Advanced Encryption Standard). AES es un método criptográfico simétrico, es decir, que usa la misma clave para cifrar y descifrar mensajes. Remitente y destinatario deben conocer la clave para poder comunicarse.

4.2.3.8. Perfiles

Ya que Zigbee está pensado para la comunicación entre diversos dispositivos, posiblemente de fabricantes diferentes, es necesario un mecanismo para hacer compatibles los mensajes, comandos, etc. que pueden enviarse unos a otros. Para ello existen los perfiles de Zigbee.

Los perfiles son la clave para la comunicación entre dispositivos Zigbee. Definen los métodos de comunicación, el tipo de mensajes a utilizar, los comandos disponibles y las respuestas que permiten a dispositivos separados comunicarse para crear una aplicación distribuida. Un perfil es un conjunto de reglas que confieren funcionalidad e interoperabilidad en el nivel de aplicación y red. Casi todo tipo de operaciones han de estar definidas en un perfil. Por ejemplo, las tareas típicas de unirse a una red o descubrir dispositivos y servicios están soportadas por el «perfil de dispositivos» Zigbee. Pueden ser públicos o privados. Los perfiles públicos son creados por la Zigbee Alliance para favorecer la interoperabilidad, mientras que los privados, para uso en sistemas que requieren aplicaciones más concretas, los diseñan las empresas o el propio cliente. Cada perfil debe tener un identificador y, obviamente, éste debe ser único. Por ello, la Zigbee Alliance se reserva el derecho de asignar identificadores a los diversos perfiles. Si es necesaria la creación de un nuevo perfil, tiene que hacerse la petición a la Zigbee Alliance. Para garantizar la comunicación entre nodos, es obligatorio que utilicen los mismos perfiles.

A continuación (Figura 15) se muestran los perfiles existentes.

PERFIL 0	Perfil privado que permite al usuario desarrollar sus aplicaciones
PERFIL 1	Home Control (HC)
PERFIL 2	Commercial Building Automation (CBA)
PERFIL 3-15	Reservados para definir futuras aplicaciones

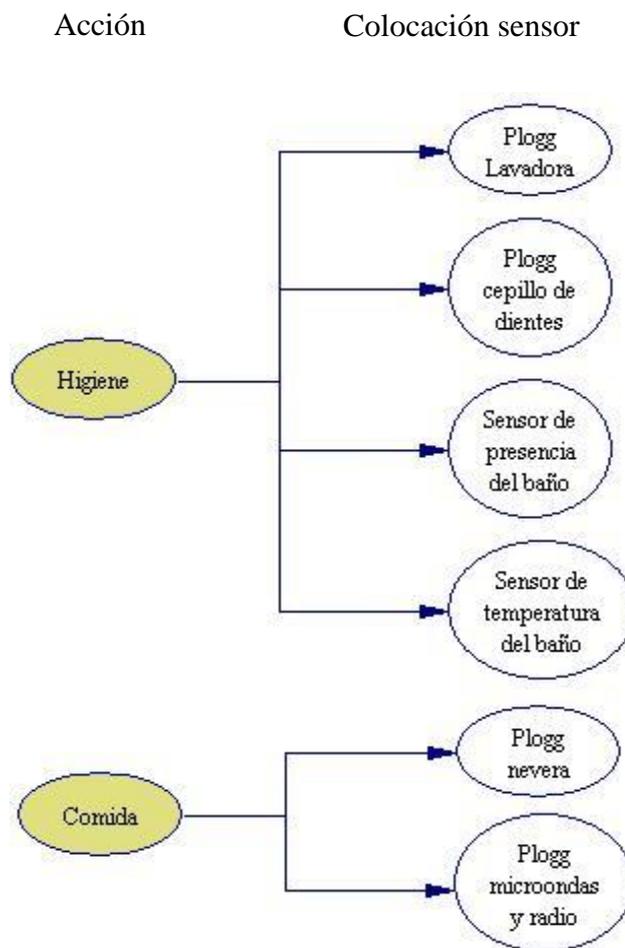
Figura 15. Perfiles Zigbee

4.2.4. Instalación de los sensores:

Para la realización del proyecto se dispuso de siete Ploggs y dos multisensores, por lo que se ha tenido que diseñar la ubicación en el hogar de los distintos sistemas para el reconocimiento de actividades.

Teniendo en cuenta el método de Katz, las actividades que se decidieron observar para este trabajo han sido: higiene, comida, ocio, salida/entrada de casa, ir al baño y el descanso.

A continuación, se recoge un esquema de las acciones relacionadas con la colocación de los sensores en el hogar (Figura 16).



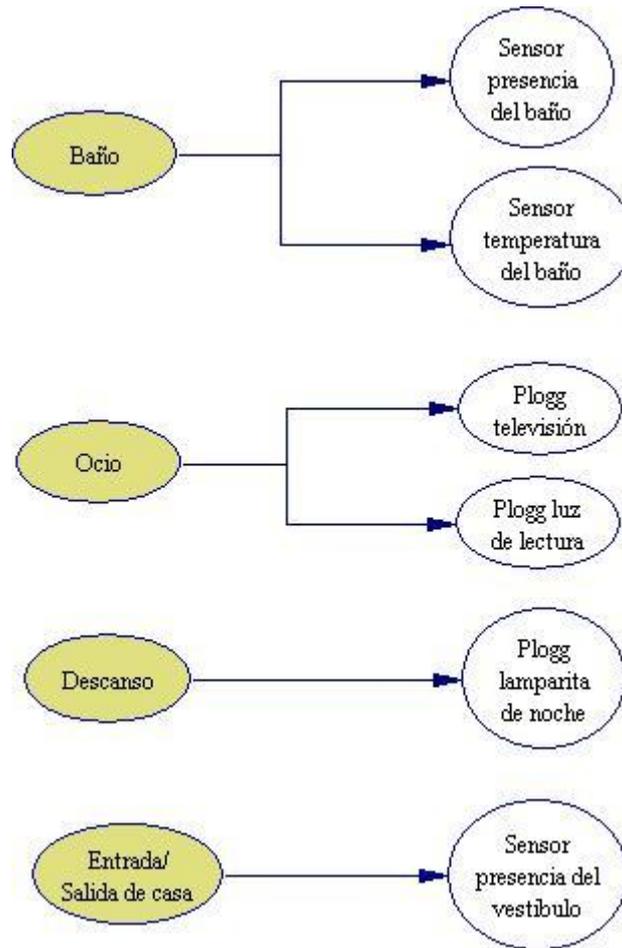


Figura 16. Actividades y sensores

Para poder detectar las acciones relacionadas con la higiene, se conectó un Plogg a la lavadora, de esta manera se consiguió saber la frecuencia de lavado de la ropa. También se conectó un detector de movimiento y temperatura en el baño además de un Plogg conectado al cepillo de dientes eléctrico.

Para detectar cuando la persona realizaba su comida, se colocó un Plogg en la nevera, dado que es el lugar donde se guardan los alimentos. Otro Plogg se incorporó a una regleta en la que estaba conectado el microondas mediante el cual el usuario calentaba los alimentos y una radio que siempre encendía para mantenerse informado mientras comía.

Para recoger los datos de las actividades relacionadas con el ocio, se conectó un Plogg a la televisión y otro a una luz de lectura que tenía en el salón.

Para detectar las entradas y salidas del hogar se situó un detector de presencia junto con uno de temperatura en el vestíbulo de la casa.

Cuando la persona entraba al baño, se quería registrar la frecuencia con la que acudía al WC aprovechando los sensores ya instalados para detectar las acciones relacionadas con la higiene.

Para detectar el descanso se colocó un Plogg en la lamparita de noche, así se consiguió saber cuándo la persona se acostaba y cuando se levantaba.

En el siguiente dibujo se muestra la colocación de los distintos sensores en el hogar (Figura 17)

- 1.- Plogg conectado al cepillo de dientes.
- 2.- Plogg conectado a la lamparita de noche.
- 3.- Plogg conectado a la luz de lectura.
- 4.- Plogg conectado a la televisión.
- 5.- Plogg conectado al microondas y la radio.
- 6.- Plogg conectado a la nevera.
- 7.- Plogg conectado a la lavadora.
- M1.- Multisensor conectado en la entrada de la casa.
- M2.- Multisensor conectado en el baño.

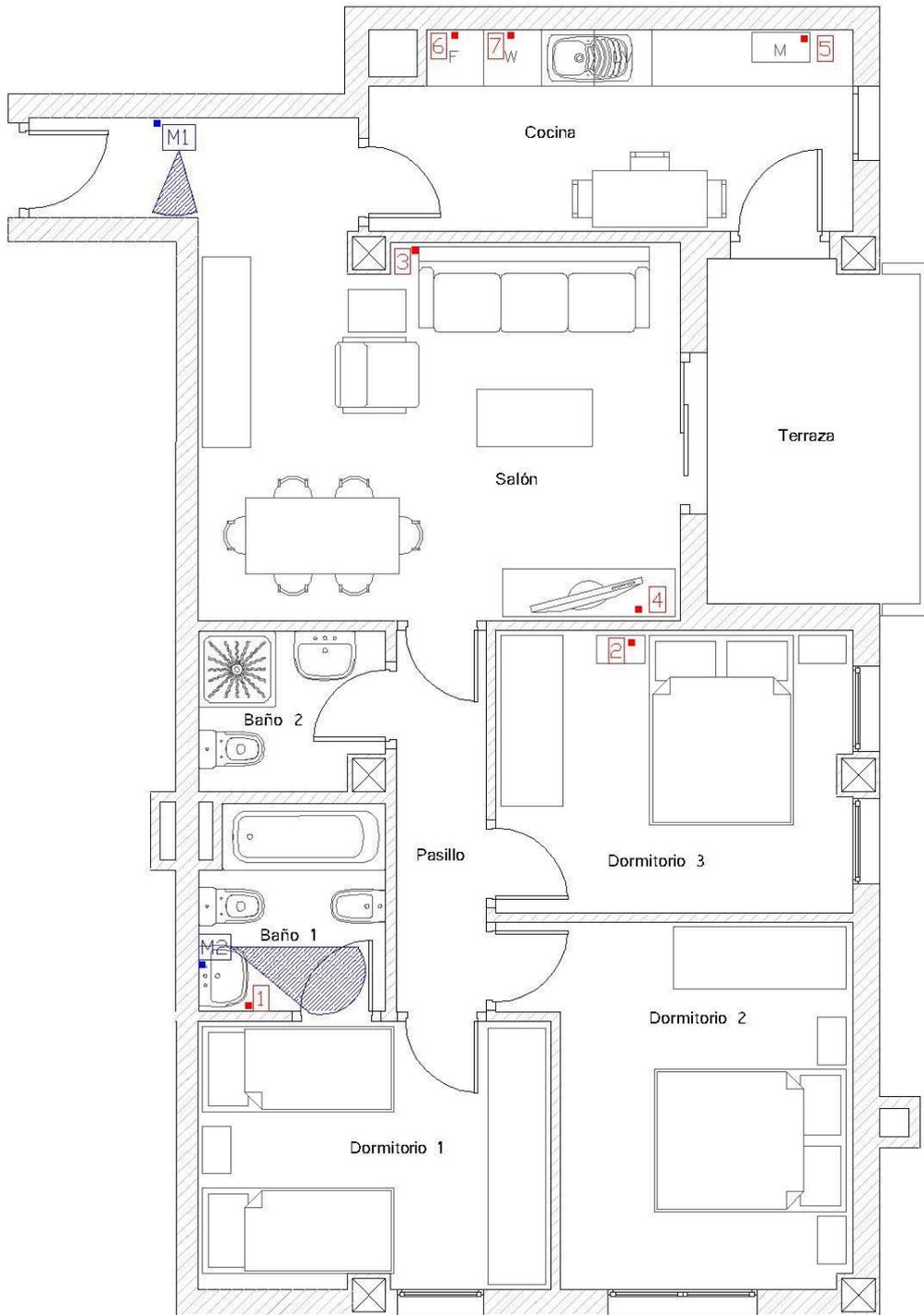


Figura 17. Distribución de los sensores en el piso

4.3 Software:

En este apartado, se han descrito los Scripts utilizados para alcanzar el objetivo planteado de reconocimiento de actividad. Para ello, en primer lugar, se han diseñaron los scripts de recogida de datos y posteriormente, los de procesamiento de actividad.

4.3.1. Script de recogida de datos:

En el caso del Script de los multisensores, se utilizó una aplicación en java previamente desarrollada por Tecnodiscap. Esta aplicación generaba un archivo con extensión *.txt con información de la temperatura y de la presencia de la persona en el hogar. En lo referente a la temperatura, la recogida de los datos mediante los sensores se realizó de forma periódica es decir, nos la registraba cada cierto periodo de tiempo. Mientras que en el caso de la posición, sólo cuando detectaba a la persona es cuando recogía la información (Figura 18).

```

C:\WINDOWS\system32\cmd.exe
INFO: ZigbeeGateway: CONNECTED TO COM5
13-nov-2011 13:09:27 es.unizar.tecnodiscap.io.zigbee.at.ZigbeeGatewayATv3 messageReceived
INFO: ZGv3: Zigbee device found in port (COM5) with eui64 (0021ED0000036232)
13-nov-2011 13:09:27 es.unizar.tecnodiscap.io.zigbee.at.ZigbeeGatewayATv3 messageReceived
INFO: ZGv3: Zigbee device connected (0021ED0000036232) Type requested (OK)
13-nov-2011 13:09:27 es.unizar.tecnodiscap.io.zigbee.at.ZigbeeGatewayATv3 messageReceived
INFO: ZGv3: Zigbee device Coordinator connected (<+N=C00,26,03,1405,01347B83A3D44BEE>)
node:0021ED0000033A12 added. Type: 4
node:0021ED00000463D7 added. Type: 4
MnDv: New message from the node 0021ED00000463D7
1321186177390;Sun Nov 13 13:09:37 CET 2011;0021ED00000463D7;18;PIR EVENT; ;
MnDv: New message from the node 0021ED0000033A12
1321186178250;Sun Nov 13 13:09:38 CET 2011;0021ED0000033A12;18;PIR EVENT; ;
MnDv: New message from the node 0021ED00000463D7
1321186185875;Sun Nov 13 13:09:45 CET 2011;0021ED00000463D7;18;PIR EVENT; ;
MnDv: New message from the node 0021ED00000463D7
1321186204828;Sun Nov 13 13:10:04 CET 2011;0021ED00000463D7;18;PIR EVENT; ;
MnDv: New message from the node 0021ED0000033A12
1321186207796;Sun Nov 13 13:10:07 CET 2011;0021ED0000033A12;17; TEMP; 23.0;
MnDv: New message from the node 0021ED00000463D7
1321186228656;Sun Nov 13 13:10:28 CET 2011;0021ED00000463D7;17; TEMP; 23.5;
MnDv: New message from the node 0021ED00000463D7
1321186235812;Sun Nov 13 13:10:35 CET 2011;0021ED00000463D7;18;PIR EVENT; ;
MnDv: New message from the node 0021ED00000463D7
1321186243046;Sun Nov 13 13:10:43 CET 2011;0021ED00000463D7;18;PIR EVENT; ;
MnDv: New message from the node 0021ED0000033A12
1321186244906;Sun Nov 13 13:10:44 CET 2011;0021ED0000033A12;18;PIR EVENT; ;
MnDv: New message from the node 0021ED00000463D7
1321186324234;Sun Nov 13 13:12:04 CET 2011;0021ED00000463D7;18;PIR EVENT; ;
MnDv: New message from the node 0021ED00000463D7
1321186385640;Sun Nov 13 13:13:05 CET 2011;0021ED00000463D7;18;PIR EVENT; ;
MnDv: New message from the node 0021ED0000033A12
1321186386046;Sun Nov 13 13:13:06 CET 2011;0021ED0000033A12;18;PIR EVENT; ;
MnDv: New message from the node 0021ED0000033A12
1321186508515;Sun Nov 13 13:15:08 CET 2011;0021ED0000033A12;17; TEMP; 23.5;
MnDv: New message from the node 0021ED00000463D7
1321186510484;Sun Nov 13 13:15:10 CET 2011;0021ED00000463D7;18;PIR EVENT; ;
MnDv: New message from the node 0021ED0000033A12
1321186510781;Sun Nov 13 13:15:10 CET 2011;0021ED0000033A12;18;PIR EVENT; ;
MnDv: New message from the node 0021ED00000463D7
1321186527703;Sun Nov 13 13:15:27 CET 2011;0021ED00000463D7;17; TEMP; 23.5;

```

Figura 18. Recogida de datos de los multisensores

En cuanto al Script de los Ploggs, se tuvo que desarrollar uno. Como es un prototipo se decidió que fuese Matlab la herramienta utilizada para ello ya que es un lenguaje que permite centrarse en el diseño más que en la forma.

4.3.2. Desarrollo del Script de los Ploggs:

Para el desarrollo del Script de recogida de datos de los Ploggs, se estaba utilizando hasta el momento, un software externo que funcionaba en Linux. Dicho software era cerrado y no permitía hacer cambios en su código, por lo que no era muy funcional. Por lo cual era muy interesante crear nuestro propio Script de recogida de datos.

Para desarrollar el Script se estudió un prototipo previo realizado en Matlab. Este Script recogía los datos de unos Ploggs que estaban predeterminados en el código y

generaba un archivo. El código elegía el Plogg al que pedía la información de una forma muy compleja y poco optimizada. Además, no se podía modificar el número de Ploggs. El principal inconveniente que tenía era su falta de robustez, cada vez que fallaba una comunicación, un Plogg se desconectaba, o si no estaba alguno de los Ploggs conectado, el programa se colgaba, lo cual era algo inaceptable debido a la utilidad que iba a tener el Script. Otro problema que tenía el programa era que generaba vectores infinitos, los cuales acababan saturando la memoria RAM del ordenador en caso de estar mucho tiempo funcionando de manera continua.

Dado que interesaba un programa robusto, que pudiera estar funcionando 24 horas al día durante muchos días seguidos, sin que tuviera problemas (debidos a que el usuario conectara o desconectara un Plogg, que el dato vía Zigbee llegara o no correctamente o que el ordenador no hiciera correctamente la comunicación serie entre el programa y el dispositivo Zigbee), decidí que tenía que hacer un programa completamente nuevo. Pude aprovechar la parte del código que traducía los datos recibidos vía Zigbee a datos de corriente, potencia, voltaje, etc. que eran los que me interesaba almacenar. Aunque tuve que realizar las adaptaciones necesarias para que funcionara con mi código.

A continuación se muestra como era el proceso de recogida de cada dato (Figura 19).

...Esperando datos de Plogg...	...Esperando datos de Plogg...
1	2
Antes counter++	Antes counter++
Despues counter++	Despues counter++
<	<
Recibidos Datos Plogg1	Recibidos Datos Plogg1
17	17
<	<
Recibidos Datos Plogg1	Recibidos Datos Plogg1
112	112
FIN Recibidos Datos Plogg1	FIN Recibidos Datos Plogg1
Estoy en cojo datos	Estoy en cojo datos
Estoy en store_Data	Estoy en store_Data
Despues counter=counter1	Despues counter=counter1
1	2
54	54
30	30
101110000001010000011100011111	101110000001010000011100011000
1	2

Figura 19. Proceso de recogida de datos de los Ploggs

4.3.2.1. Funcionamiento del programa:

El programa diseñado almacenaba en ficheros de texto la información de consumos recogida por los Ploggs en cada instante. Para cada Plogg se generaba un fichero de texto (Figura 20). Además, cada día se creaba de cada Plogg un nuevo fichero de texto.

```

Año Mes Dia Hora Minuto Segundo Watts CumWattsCon(Kwh) Frequency(Hz) RMSVolts(V) RMSCurr(A) PloggOnTime(Sg) EquipOnTime
1
11 10 11 18 41 14 1.040000 6.246000 50.000000 233.987000 0.017000 25689 25325
2
11 10 11 18 41 25 1.040000 6.246000 50.000000 233.798000 0.017000 26788 26424
3
11 10 11 18 41 34 1.040000 6.246000 50.000000 233.718000 0.017000 27675 27311
4
11 10 11 18 41 43 1.040000 6.246000 50.000000 233.582000 0.017000 28579 28214
5
11 10 11 18 41 52 1.040000 6.246000 50.000000 233.437000 0.017000 29473 29109
6
11 10 11 18 42 1 1.040000 6.246000 50.000000 233.840000 0.017000 30385 30021
7
11 10 11 18 42 10 1.040000 6.246000 50.000000 233.940000 0.017000 31275 30911
8
11 10 11 18 42 19 1.040000 6.246000 50.000000 233.663000 0.017000 32175 31811
9
11 10 11 18 42 28 1.040000 6.246000 50.000000 233.738000 0.017000 33072 32708
10
11 10 11 18 42 37 1.040000 6.246000 50.000000 233.364000 0.017000 33972 33608
11
11 10 11 18 42 46 1.040000 6.246000 50.000000 233.259000 0.017000 34824 34503
12
11 10 11 18 42 55 1.040000 6.246000 50.000000 233.448000 0.017000 35768 35404
13
11 10 11 18 43 4 1.040000 6.246000 50.000000 233.047000 0.017000 36670 36306
14
11 10 11 18 43 13 1.040000 6.246000 50.000000 233.009000 0.017000 37568 37204
15
11 10 11 18 43 22 1.040000 6.246000 50.000000 233.072000 0.017000 38469 38105
16
11 10 11 18 43 31 1.040000 6.246000 50.000000 232.972000 0.018000 39370 39005
17
11 10 11 18 43 40 1.040000 6.246000 50.000000 233.506000 0.017000 40267 39903
18
11 10 11 18 43 49 1.040000 6.247000 50.000000 233.365000 0.017000 41170 40806

```

Figura 20. Fichero de texto con los datos registrados de un Plogg

Los datos a configurar por primera vez al utilizar el Script fueron:

- La dirección MAC de cada Plogg que se deseaba examinar en una variable del tipo estructura (Figura 21).
- Una constante llamada “MAXPLOGG” en la que estaba almacenado el número de Ploggs que había en total.
- Una variable llamada “puerto” que guardaba el puerto serie en el que estaba conectado el dispositivo de la red Zigbee.

```

Plogg(1).mac = '000D6F000026C7ED';
Plogg(2).mac = '000D6F000026BCC9';
Plogg(3).mac = '000D6F0000353F22';
Plogg(4).mac = '000D6F000026C7B9';
Plogg(5).mac = '000D6F00007D4COE';
Plogg(6).mac = '000D6F0000354380';
Plogg(7).mac = '000D6F000026C892';

```

Figura 21. Estructura de las direcciones MAC

El manejo del controlador Zigbee se hizo mediante la generación de un puerto serie estándar virtual. Así los datos llegaban como si vinieran por un puerto COM serie.

El programa, en primer lugar, abría el puerto que estaba configurado (Figura 22). Al abrir el puerto también se habilitaba una función de Matlab llamada “BytesAvailableFcn” que permitía que se generase un evento cuando se recibían datos por el puerto serie. A continuación, daba la opción al usuario de si quería iniciar la recogida de datos, o por el contrario, deseaba salir del programa, ya que la recogida de datos era un bucle infinito, en el cual se pedía constantemente datos a los Ploggs uno detrás de otro.

```

184  function Abrir_Puerto(ComPort)
185  -      s=serial(ComPort);
186  -      set(s, 'BaudRate', 115200);
187  -      set(s, 'Terminator', 'CR');
188  -      set(s, 'Timeout', 0.4);
189  -      set(s, 'ReadAsyncMode', 'continuous');
190  -      s.BytesAvailableFcn = (@mydispcallback);
191  -      s.ErrorFcn = (@mierrorcallback);
192  -      fopen(s);
193  -      disp('puerto abierto')
194  -  end

```

Figura 22. Función de abrir el puerto COM

Una vez que el usuario iniciaba la recogida de datos, el programa inicializaba las variables e iba a la función de “recogida_Datos” el cual era el bucle infinito.

En “recogida_Datos” el programa comprobaba si la recepción de datos había finalizado o estaba pendiente, y si la recepción había fallado o continuaba esperando dato. Si la recepción había terminado o fallado, el programa pasaba al siguiente Plogg. Para eso, inicializaba las variables e iba a la función “guarda_Datos” que pedía el dato al siguiente Plogg. Si la recepción no había fallado o no había terminado, comprobaba si el dato que estaba llegando era correcto y esperaba hasta completar ese dato. En caso de que el dato fuese incorrecto daba recepción fallida (Figura 23) y pedía otro dato al siguiente Plogg. La espera del dato correcto se hacía mediante un evento que se

generaba asíncronamente cuando llegaban datos al puerto serie. Ésta era la función “mydispcallback”.

```
149 - tam = size(M_Data_All_1);
150 - if (tam(2) == 28)
151 -     recepcionFallada = 1;
152 -     M_Data_All_2 = [];
153 -     M_Data_All_1 = M_Data_All_2;
154 - end
155
156 - if(toc(Tiempo1) > 8)
157 -     recepcionFallada = 1;
158 -     M_Data_All_2 = [];
159 -     M_Data_All_1 = M_Data_All_2;
160 - end
```

Figura 23. Control de si el dato falla durante la comunicación Zigbee

La función “guarda_Datos” pedía un dato al Plogg mediante el puerto serie. Si el dato se había pedido correctamente (Figura 24), esperaba la llegada del dato en la función “recogida_Datos”. Por el contrario, si había algún fallo durante la comunicación con el puerto serie, lo detectaba y lo cerraba. El puerto serie lo tenía que cerrar porque cuando había un fallo se deshabilitaba el evento asíncrono que se generaba cuando llegaban datos al puerto serie. Una vez cerrado, el programa inicializaba los vectores de recogida de datos, activaba la variable “recepcionFallada”, abría otra vez el puerto serie e iba a la función “recogida_Datos”.

```
205 -     recepcionTerminada = 0;
206 -     counter(numlogg) = counter(numlogg) + 1;
207 -     try
208 -         fprintf(s, '%s\n', Plogg(numlogg).orden);
209 -     catch err
210 -         if (strcmp(err.identifier, 'MATLAB:serial:fprintf:opfailed'))
211 -             msg = sprintf('%s', 'FALLO', 'FALLO', 'FALLO');
212 -             Cerrar_Puerto();
213 -             recepcionFallada = 1;
214 -             M_Data_All_2 = [];
215 -             M_Data_All_1 = M_Data_All_2;
216 -
217 -             Abrir_Puerto(puerto);
218 -
219 -         else
220 -             rethrow(err);
221 -         end
222 -     end
```

Figura 24. Control de fallos en el puerto serie

La función “mydispcallback” (Figura 25) se activaba cuando llegaban datos al puerto serie. En ella se iban almacenando en un vector los datos que llegaban del Plogg hasta que el dato estaba completo. Una vez conseguido esto, se procesaba para que quedaran sólo la parte que nos interesaba. La cual se convertía a voltaje, potencia, etc. para su posterior almacenamiento en un fichero de texto. Este fichero se creaba nuevo cada día para cada Plogg. Una vez terminado este proceso se pasaba a pedir el dato al siguiente Plogg.

```

234 function mydispcallback(obj, event)
235 -     if (obj.BytesAvailable > 0),
236 -         nData = obj.BytesAvailable;
237 -         disp('<');
238 -         disp('Recibidos Datos Plogg1');
239 -         temp = fread(obj,nData);
240 -         M_Data_All_1 = [M_Data_All_1 temp'];
241
242 -         tam = size(M_Data_All_1);
243 -         disp(tam(2));
244 -         if (tam(2) == 112)
245 -             disp('FIN Recibidos Datos Plogg1');
246 -             recepcionTerminada = 1;
247 -             DatoBruto = M_Data_All_1;
248 -             cojoDatos();
249 -         end
250 -     end
251 - end

```

Figura 25. Función mydispcallback que se activa con la llegada de datos al puerto serie

4.3.2.2. Diagrama de Flujo del Script de recogida de datos de los Ploggs:

En las siguientes figuras se muestran el diagrama de flujo principal de la aplicación (Figura 26) y el diagrama de flujo de la interrupción de la recogida del dato (Figura 27).

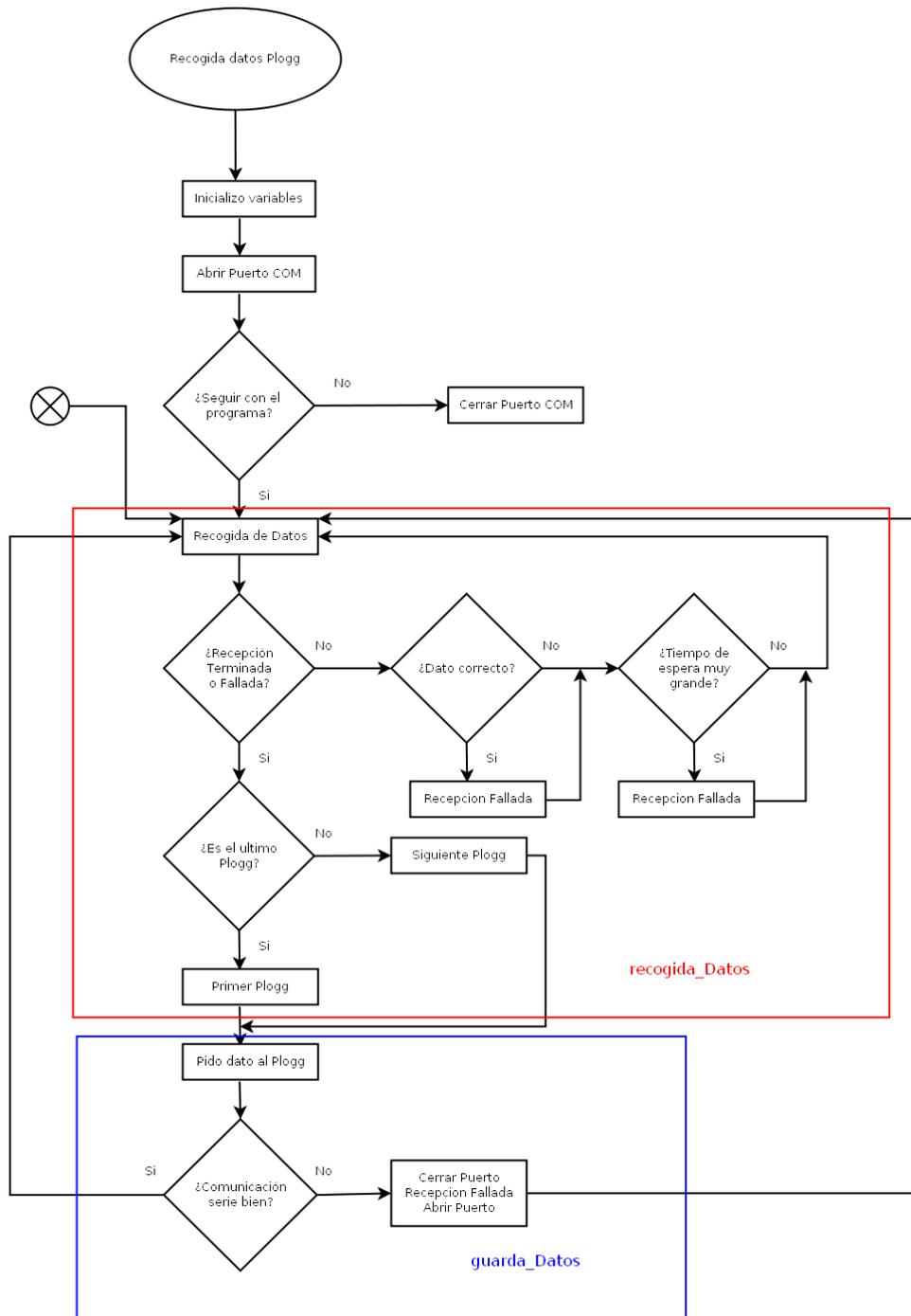


Figura 26. Diagrama de flujo principal

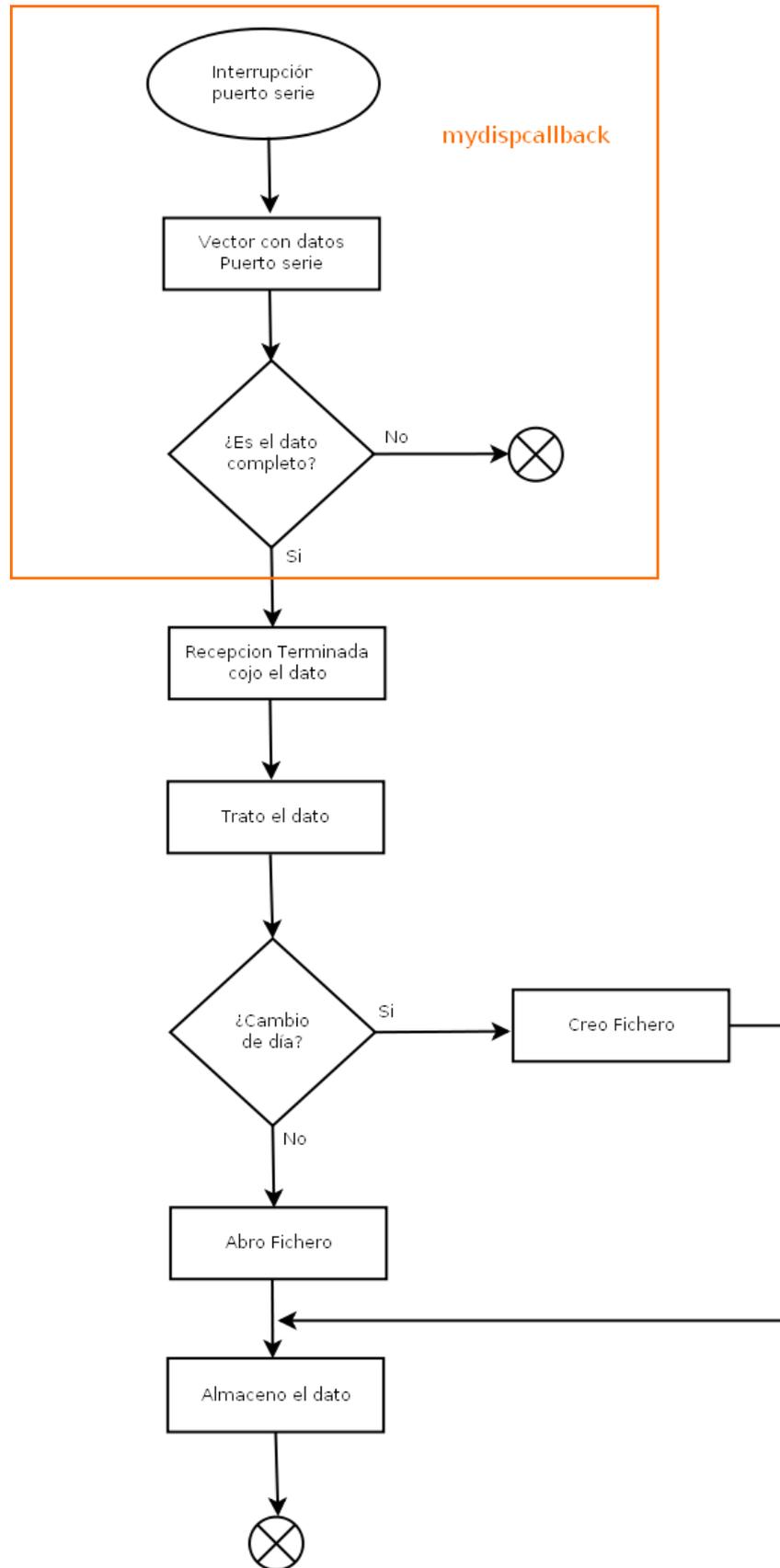


Figura 27. Diagrama de flujo de la interrupción

4.3.2.3. Dificultades añadidas durante el desarrollo de la aplicación:

A lo largo del diseño del programa principal necesario para la realización de mi proyecto me encontré con distintos problemas que tuve que solucionar, algunos de los cuales quiero destacar en esta memoria. Por eso, los he expuesto a continuación:

- Empecé a trabajar con un script de partida muy estático que recogía sólo los datos de 5 Plogg. De tal manera, que para cada Plogg que utilizaba, se tenía que copiar todo el código. Yo quería que mi programa independientemente del número de Ploggs, funcionara sin la necesidad de copiar el código. Para ello, pensé que la mejor manera era crear una estructura de datos con la dirección MAC de cada Plogg y una variable que sería el número de Plogg. De esta forma, era muy fácil añadir o quitar Plogg al sistema con sólo cambiarlos al principio.
- Durante el proceso de recepción de datos, el programa se quedaba colgado porque no tenía en cuenta ni los datos que llegaban erróneos, ni los que no llegaban. Para solucionar este problema, se decidió controlar el tiempo de espera de un dato y se determinó que, si éste era mayor que 8 segundos se daba como dato erróneo. Pero esto ralentizaba mucho el programa. Además, se observó que para la mayoría de fallos de datos erróneos se recibía el mismo código por lo que se optó por controlar los datos con código erróneo y mantener el control del tiempo de espera (8 segundos) por si hubiera otro tipo de error que hiciera que el programa se quedara colgado.
- Una vez que tenía ya el programa funcionando, inicié un periodo de pruebas durante bastante tiempo (24-48 horas) de forma continua. Durante estas pruebas, surgió un error que se generaba sin una lógica aparente, por el cual, el programa dejaba de funcionar. Este error podía aparecer tanto a la hora de empezar a funcionar como a las 20 ó 30 horas, por lo que el programa no tenía la robustez exigida para los objetivos que se habían planteado. Tras estudiar a fondo el fallo detecté que se trataba de un error durante el proceso de comunicación del puerto serie COM por sobrepasar el tiempo de espera que estaba configurado. Para solucionarlo, fue necesario crear un try-catch, en el cual, en caso de fallo, se

cerraba el puerto y se volvía a abrir y consideraba el dato como erróneo e inicializaba las variables.

4.3.3. Script de análisis de datos:

Para realizar el análisis de los datos se desarrollaron dos Scripts en Matlab. Uno, leía el fichero de texto de los datos del multisensor. Los pasaba a una matriz de cuatro dimensiones en las cuales estaba: el día, el sensor, el tipo de dato (si es temperatura o presencia) y el valor (Figura 28). En total se almacenaban 1440 datos cada día que correspondían a los minutos que hay en un día. Se decidió que una buena manera de saber la actividad de la persona era minuto a minuto porque teníamos una resolución suficiente para saber si había actividad o no en ese periodo de tiempo. Si hubiese sido segundo a segundo se hubiesen recogido muchos datos que no serían relevantes para la deducción de actividad. Mientras que si hubiese sido hora a hora se hubiesen perdido muchos datos importantes.

```
35 -     if (str2num(b{3})==17)
36 -         tiposensor=1;
37 -         numminuto=str2num(c{1}) * 60 + str2num(c{2})+1;
38 -         datosMulti(numdia,numsensor,tiposensor,numminuto)=str2num(d{1});
39 -     end
40 -     if (str2num(b{3})==18)
41 -         tiposensor=2;
42 -         numminuto=str2num(c{1}) * 60 + str2num(c{2})+1;
43 -         datosMulti(numdia,numsensor,tiposensor,numminuto)=1;
44 -     end
```

Figura 28. Vector con el dato del multisensor

El otro Script, leía los ficheros de texto de los Ploggs. Dentro del Script se trataba a cada Plogg de manera independiente, ya que para detectar actividad o no en el Plogg, se miraba la potencia o la corriente, dependiendo del aparato que estaba conectado al Plogg, lo que generaba márgenes distintos entre unos valores u otros.

En los diagramas de flujo que hay a continuación (Figura 29, 30, y 31) se detalla mejor el funcionamiento de estos dos scripts. Se creó algún script más para dibujar las gráficas o poner los datos de forma que se interpretaran más fácilmente pero no añado aquí una descripción de estos aunque los códigos figuran en los anexos.

4.3.3.1. Diagrama de Flujo del Script de tratamiento de datos del multisensor:

En este diagrama (Figura 29) se describe la forma de almacenar los datos de la actividad en un vector. En primer lugar, se abre el archivo y lee una tira de datos. Si no era el final del archivo, lee otra tira de datos. Para eso, deducía el día al que pertenecían esos datos y determinaban de cual de los dos multisensores era el dato. A continuación, averiguaba si era un dato de posición o de temperatura. En el caso de que fuese de posición, almacenaba que había habido movimiento. Mientras que si era de temperatura, almacenaba el valor de la temperatura. Una vez hecho esto, volvía a cuestionarse si era el final del archivo. Si no lo era, repetía el proceso hasta que leía todos los datos.

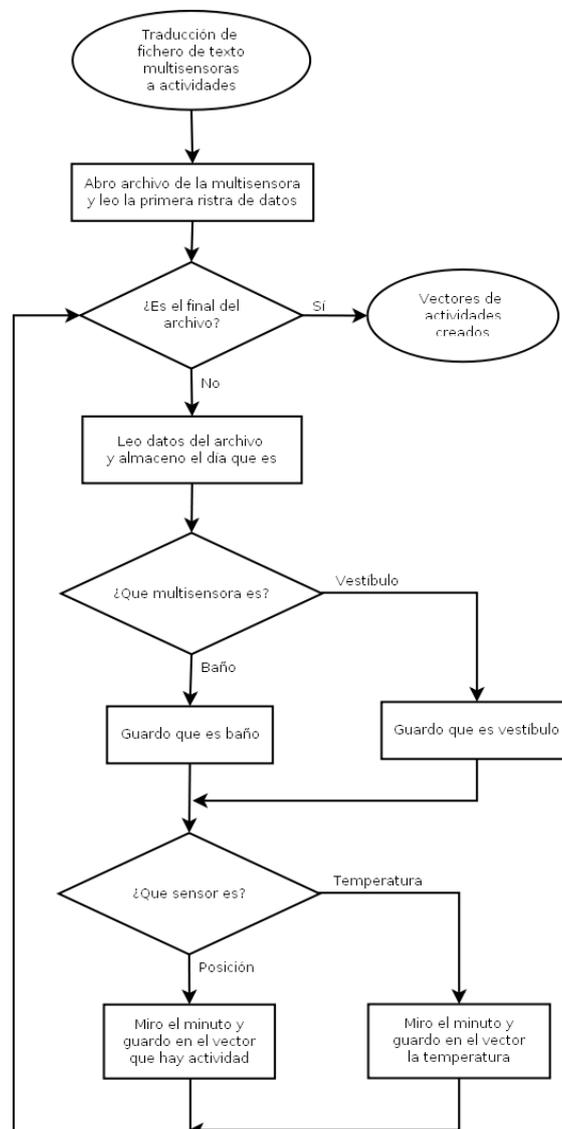


Figura 29. Diagrama Script de tratamiento de datos de los multisensores

4.3.3.2. Diagramas de Flujo del Script de tratamiento de datos de los Ploggs:

En el primer diagrama (Figura 30) se hace una descripción global del funcionamiento. En este caso, se repetía para todos los días que se habían muestreado datos. Se leía cada Plogg uno detrás de otro para generar su vector. Esto se hacía así, porque dependiendo del aparato que estaba conectado al Plogg, la lógica para deducir la actividad era diferente.

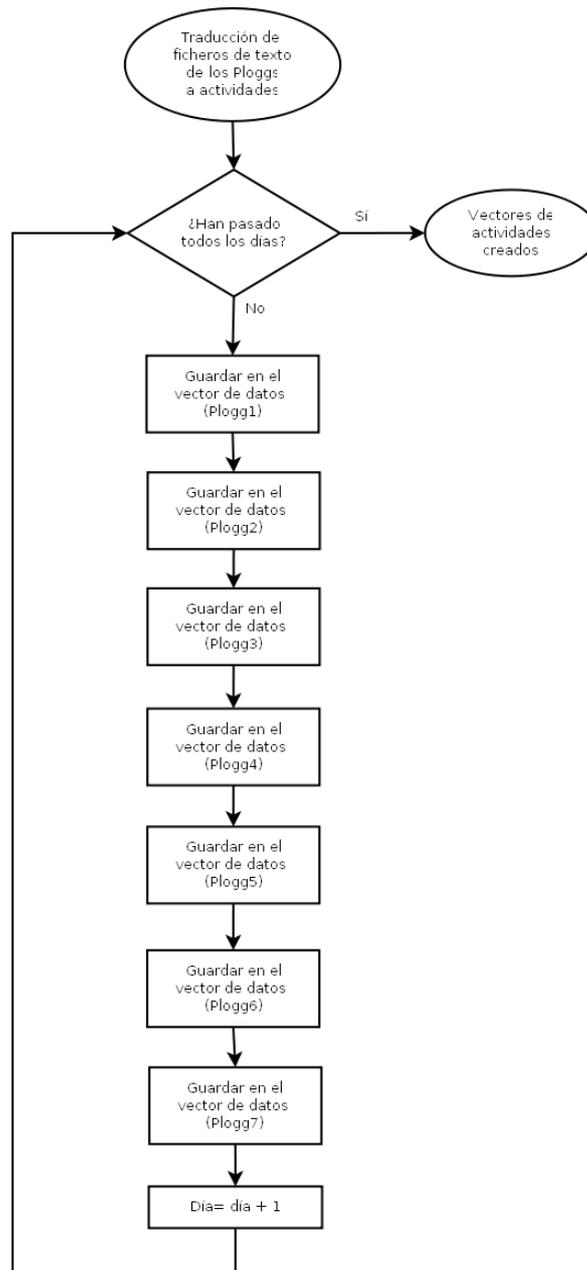


Figura 30. Diagrama global del script de tratamiento de datos de Ploggs

En el segundo diagrama (Figura 31) se especifica como se guarda el dato de cada Plogg. En primer lugar, se abría el fichero de cada Plogg. A continuación, leía el primer dato y comprobaba si era el final del archivo. Si no lo era, leía otro dato y comprobaba la condición de si había actividad en ese Plogg. El que hubiese condición dependía del aparato que estaba conectado a cada Plogg. Si esa condición decía que había habido actividad se almacenaba un uno y si no, un cero. Volvía a comprobar si era el final del archivo, en caso afirmativo se pasaba al siguiente Plogg.

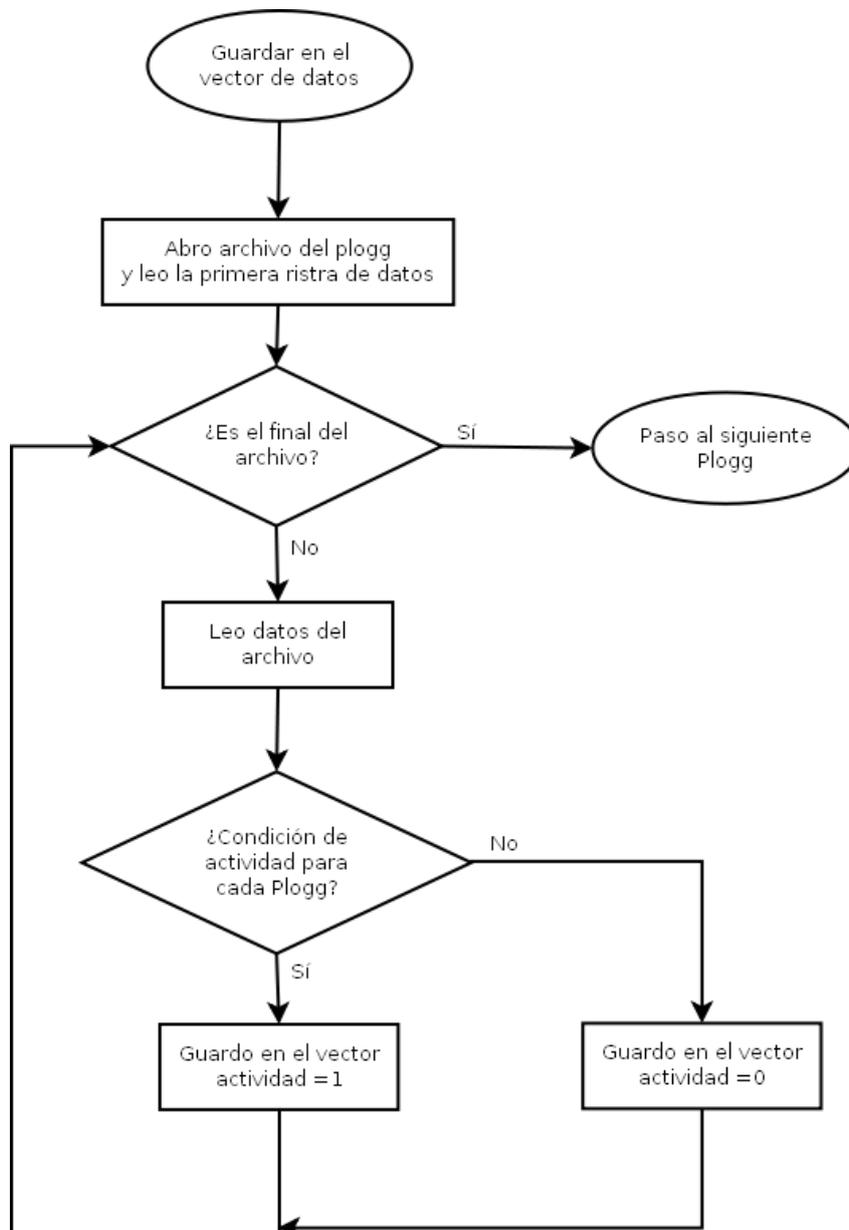


Figura 31. Diagrama de cómo se guarda el dato de un Plogg

Para la toma de decisión entre si había actividad o no en cada Plogg, se han aplicado estas condiciones:

En los casos de los Plogg de la mesita de noche, de la lamparita de lectura, de la televisión y el compartido por el microondas y la radio se consiguió diferenciar si estaba conectado o no visualizando si la potencia era cero (no conectado) u otro valor distinto de cero (conectado).

El Plogg de la nevera fue el más complejo. Se pretendía detectar que se abría la puerta por la potencia que consumía la bombilla que daba luz al interior de la nevera, pero el problema era que siempre estaba consumiendo y además, no siempre consumía el mismo valor porque este consumo dependía de si el motor estaba encendido o apagado. Para ver esa diferencia de potencia independientemente del momento en que estaba la nevera, había que encontrar un escalón cuya altura fuese la diferencia que hay entre que la bombilla estuviese apagada o encendida.

En cuanto al Plogg de la lavadora se observó que la potencia era cero pero además, se filtró un pequeño consumo residual que tiene periódicamente.

Y por último, en el Plogg del cepillo de dientes no se consiguió diferenciar entre si estaba conectado o no a la base.

4.4 Proceso de recogida de datos en el experimento:

Para realizar la recogida de datos de forma continua, durante un periodo de tiempo relativamente largo hubo que esperar al puente del Pilar. Así se pudo disponer de un piso en el que sólo viviera una persona.

El tiempo de recogida de datos fue de cinco días, de los cuales, tres fueron completos. El primer día se instalaron todos los sensores, y se puso en funcionamiento el sistema entorno a las 19:40 de la tarde y el último día se paró el sistema en torno a las 10:20 de la mañana.

Durante toda la recogida de datos, se apuntó todas las actividades que realizaba el usuario en el hogar (Tabla 4). Comparando estos datos con los que registró el sistema se pudo comprobar el correcto o incorrecto funcionamiento del prototipo.

Día 1	
Hora	Actividad
19:40	Inicio recogida de datos
19:45	Voy al baño
19:52	Salgo del baño
19:53	Abro la nevera
19:54	Me voy a la piscina
22:11	Vuelvo de la piscina
22:12	Cierro la puerta
22:13	Enciendo la televisión
22:26	Apago la televisión
22:28	Me ducho
22:40	Termino la ducha
22:47	Empiezo a cenar
23:25	Termino de cenar
23:26	Enciendo la televisión

Día 2	
Hora	Actividad
00:37	Ir al baño y lavarme los dientes
01:41	Apago la televisión
01:42	Me voy a dormir
09:12	Me levanto de dormir
09:14	Ir al baño
09:19	Empiezo a desayunar
09:32	Termino de desayunar
09:33	Me lavo los dientes
09:37	Termino de lavarme los dientes
09:53	Enciendo la televisión
10:11	Apago la televisión
10:12	Empiezo a leer
13:21	Termino de leer
13:22	Ir al baño
13:27	Salgo de casa
14:35	Vuelvo a casa
14:38	Voy a lavarme al baño
14:59	Empiezo a comer
15:38	Termino de comer
15:39	Me lavo los dientes
15:42	Termino de lavarme los dientes
15:45	Enciendo la televisión
17:45	Apago la televisión
18:14	Voy al baño
18:21	Salgo del baño
18:29	Entro a peinarme al baño
18:37	Abro la nevera
18:52	Salgo a patinar

19:56	Vuelvo de patinar
20:05	Me ducho
20:21	Termino la ducha
20:53	Empiezo a cenar
21:50	Termino de cenar
21:51	Me lavo los dientes
21:57	Termino de lavarme los dientes
21:58	Enciendo la televisión

Día 3	
Hora	Actividad
00:21	Apago la televisión
00:23	Voy al baño
00:25	Me voy a dormir
10:15	Me levanto de dormir
10:16	Voy al baño
10:20	Empiezo a desayunar
10:32	Termino de desayunar
10:33	Me lavo los dientes
10:36	Termino de lavarme los dientes
10:54	Voy al baño
11:02	Termino en el baño
11:03	Enciendo la televisión
11:09	Apago la televisión
12:18	Salgo de casa
13:22	Vuelvo a casa
13:31	Ir al baño
13:45	Empiezo a comer
14:25	Termino de comer
14:26	Me lavo los dientes
14:29	Termino de lavarme los dientes
14:31	Enciendo la televisión
16:44	Apago la televisión
16:45	Ir al baño
16:50	Me voy a la piscina
19:21	Vuelvo de la piscina
19:30	Ir al baño
19:34	Empiezo a merendar
20:07	Termino de merendar
20:10	Me doy una ducha
20:29	Termino la ducha
20:34	Salgo de casa
23:39	Creo que llego un poco antes a casa
23:49	Ir al baño
23:54	Enciendo la televisión

Día 4	
Hora	Actividad
01:20	Apago la televisión
01:21	Voy al baño y me lavo los dientes
01:34	Termino en el baño
01:36	Me voy a dormir
10:09	Me levanto de dormir
10:12	Voy al baño
10:15	Empiezo a desayunar
10:33	Termino de desayunar
10:37	Me lavo los dientes
10:44	Pongo la lavadora
10:47	Enciendo la televisión
11:45	Apago la televisión
11:46	Voy al baño
12:35	Más o menos acaba la lavadora
12:45	Salgo de casa
13:56	Vuelvo a casa
14:22	Empiezo a comer
14:27	Ir al baño
15:06	Termino de comer
15:07	Ir al baño y lavar los dientes
15:11	Enciendo la televisión
16:30	Voy al baño
17:32	Apago la televisión
17:33	Empiezo a limpiar el baño
17:45	Termino de limpiar el baño
17:47	Empiezo a merendar
18:24	Termino de merendar
18:33	Me doy una ducha
18:46	Termino la ducha
18:50	Salgo de casa
21:29	Vuelvo a casa
21:30	Ir al baño
21:35	Empiezo a cenar
22:41	Termino de cenar
22:42	Me lavo los dientes
22:44	Termino de lavarme los dientes
22:45	Enciendo la televisión
22:49	Empiezo a leer

Día 5	
Hora	Actividad
01:49	Apago la televisión
01:50	Dejo de leer
01:52	Ir al baño
01:54	Me voy a dormir
09:25	Me levanto de dormir
09:29	Ir al baño
09:31	Empiezo a desayunar
09:43	Termino de desayunar
10:00	Me lavo los dientes
10:19	Paso por el sensor de salir
10:20	Fin recogida de datos

Tabla 4. Actividades recogidas por el usuario

Este experimento ha sido una prueba para validar toda la herramienta. Una vez logrado esto, se utilizará para tomar muchos datos durante más días.

4.5 Actividades reconocidas:

Como ya he citado anteriormente en la memoria, las actividades que han sido reconocidas son las siguientes: higiene, comida, ocio, salir de casa, ir al baño y descanso.

4.5.1. Higiene:

En el caso de las actividades relacionadas con la higiene se emplearon varios Ploggs: el de la lavadora, el del cepillo de dientes y el sensor de temperatura y posición del baño.

Tras analizar los datos recogidos por los sensores, se dedujo que el Plogg colocado en el cepillo de dientes no daba ningún tipo de información ya que no se detectaba la conexión-desconexión del cepillo.

Con el sensor de posición del baño, se sabía que el usuario estaba en él, pero no se podía diferenciar qué actividad estaba realizando. Por lo que la solución, hubiese sido poner diversos sensores: uno en el lavabo, otro en el váter y un último en la ducha. Por el contrario, mediante el sensor de temperatura sí que se consiguieron resultados

interesantes, ya que cuando el individuo se duchaba, la temperatura en el baño aumentaba.

A continuación, se muestran los cuatro gráficos, uno por día que recogen la acción de ducharse (Gráficos 1, 2, 3 y 4).

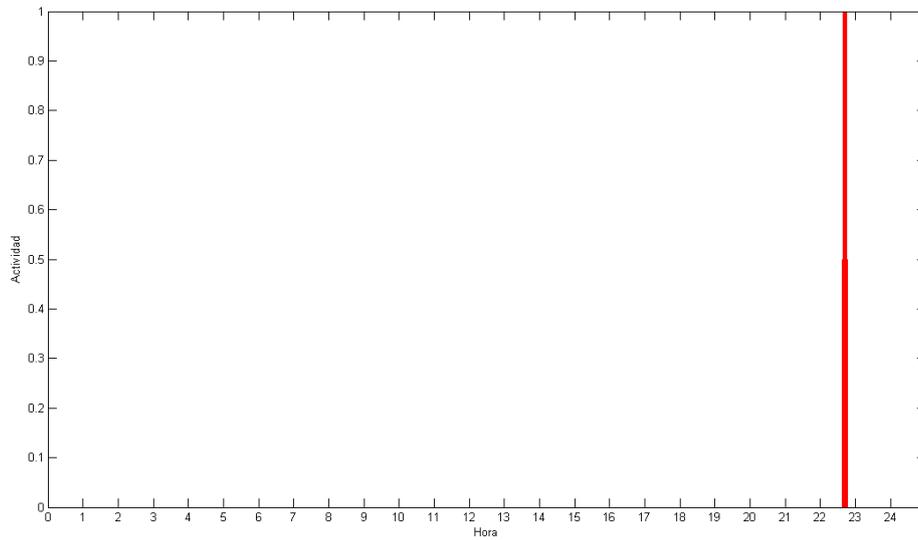


Gráfico 1. Ducha del primer día

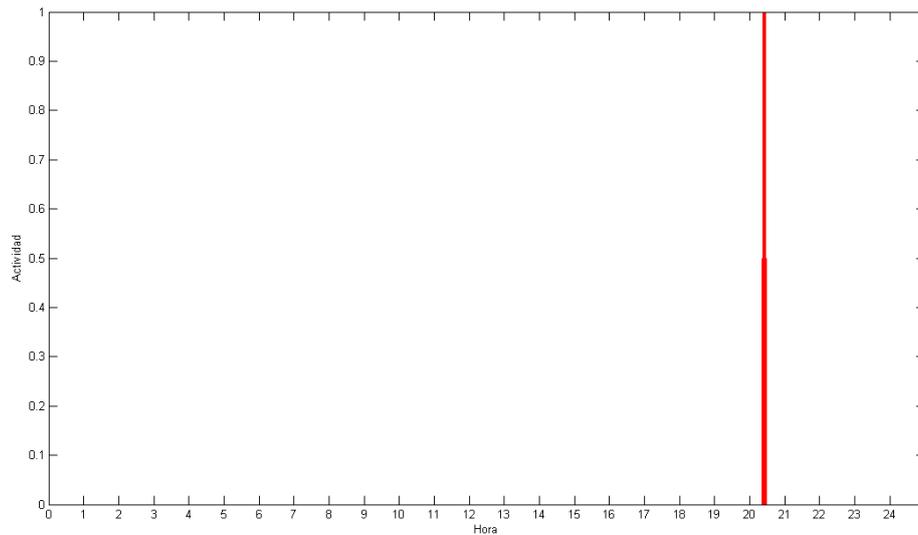


Gráfico 2. Ducha del segundo día

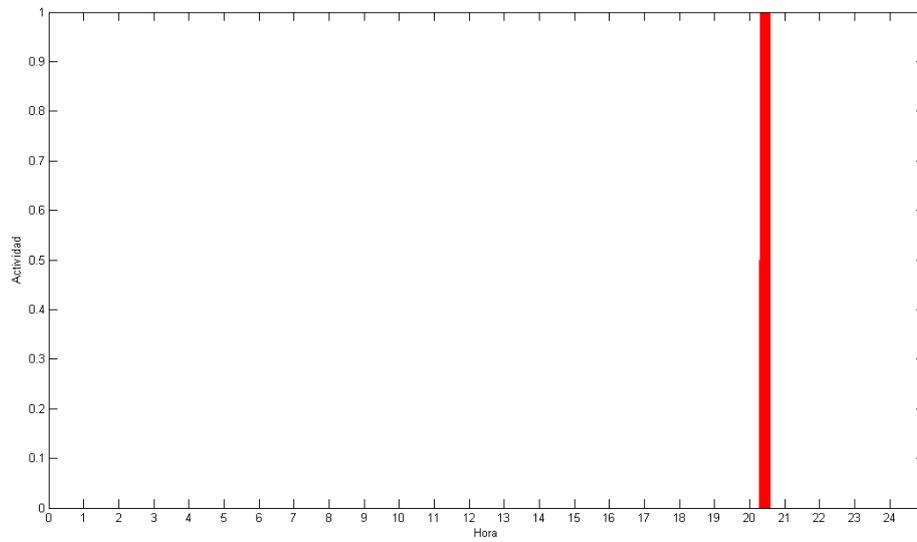


Gráfico 3. Ducha del tercer día

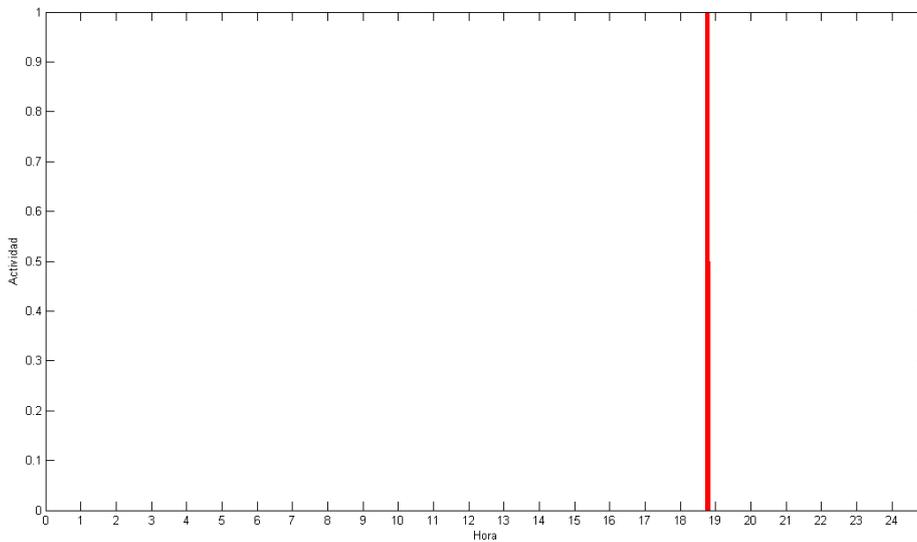


Gráfico 4. Ducha del cuarto día

El Plogg colocado en la lavadora daba información de la frecuencia con la que la persona lavaba la ropa.

En conclusión, en cuanto a las actividades relacionadas con la higiene, durante los 5 días que duró el estudio, la persona lavó la ropa 1 día y se duchó 4 días.

En los dos gráficos siguientes se muestran los resultados del Plogg de la lavadora (Gráficos 5 y 6).

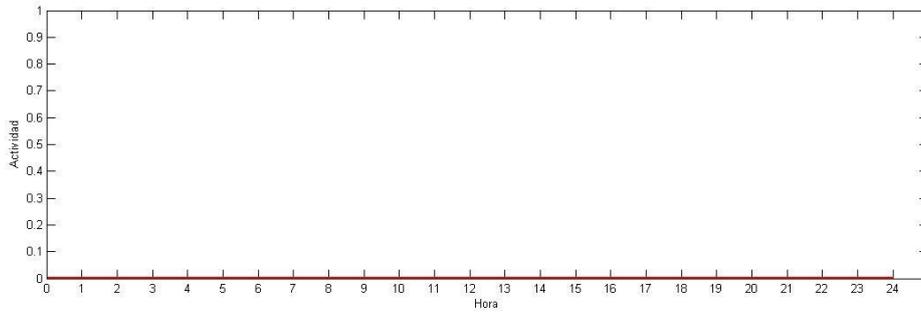


Gráfico 5. Registro de los días que no se ha puesto la lavadora.

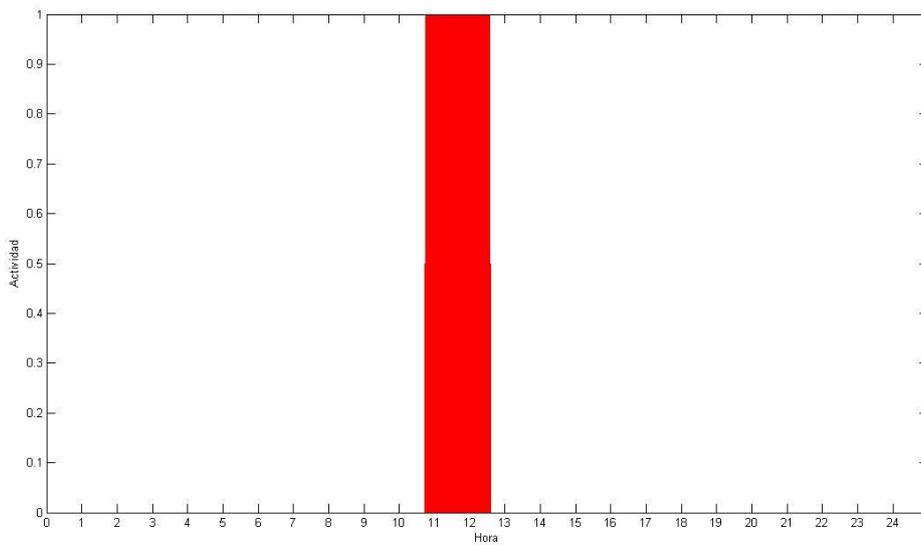


Gráfico 6. Registro del cuarto día que fue en el que se utilizó la lavadora.

4.5.2. Comida:

En el caso de la comida, se utilizaron los Plogg de la nevera y el de la radio-microondas. Como ya se ha dicho anteriormente, la persona siempre que comía conectaba la radio. De esta manera, con los datos que se obtuvieron a partir del Plogg de la radio se podía deducir los momentos que dedicaba esa persona a comer. También se observó que el Plogg de la nevera era un buen referente para detectar las comidas que hacía esa persona debido a que coincidían los datos de una mayor frecuencia de aperturas de la puerta de la nevera con los periodos de las comidas.

A continuación se muestra en los siguientes gráficos la comparativa de estos resultados (Gráficos 7, 8, 9, 10, 11, 12, 13, 14, 15, 16).

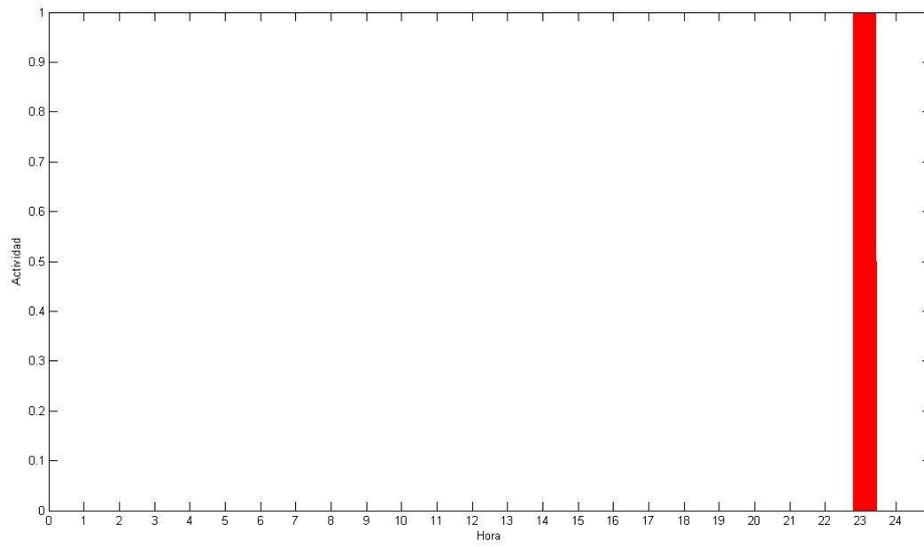


Gráfico 7. Registro de las comidas del primer día con el Plogg de la radio

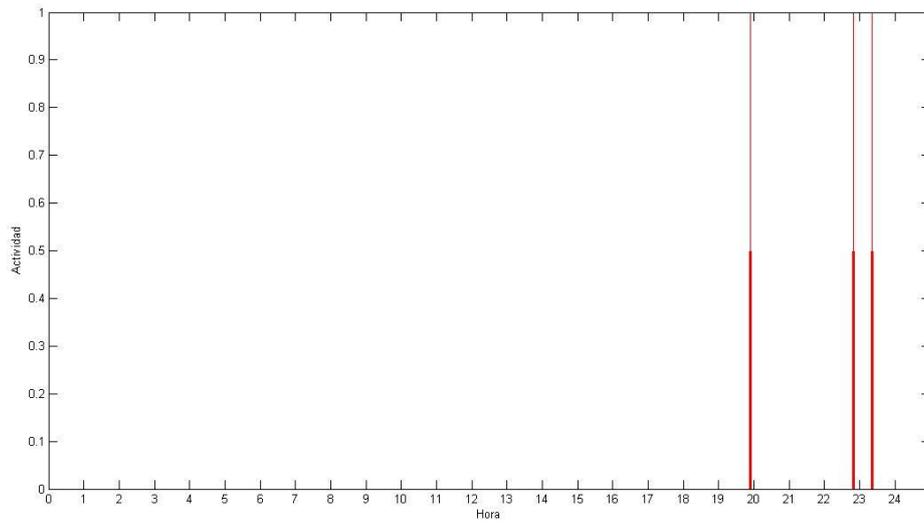


Gráfico 8. Registro de las comidas del primer día con el Plogg de la nevera

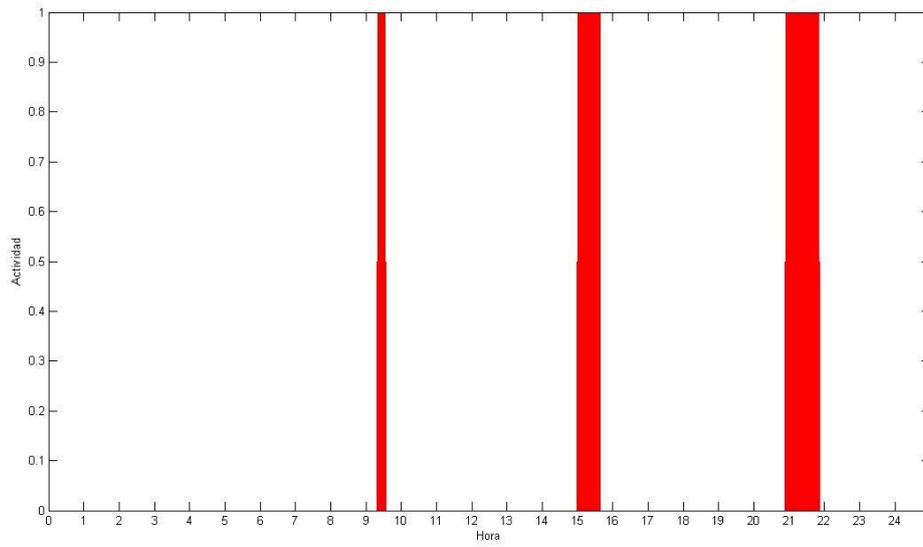


Gráfico 9. Registro de las comidas del segundo día con el Plogg de la radio

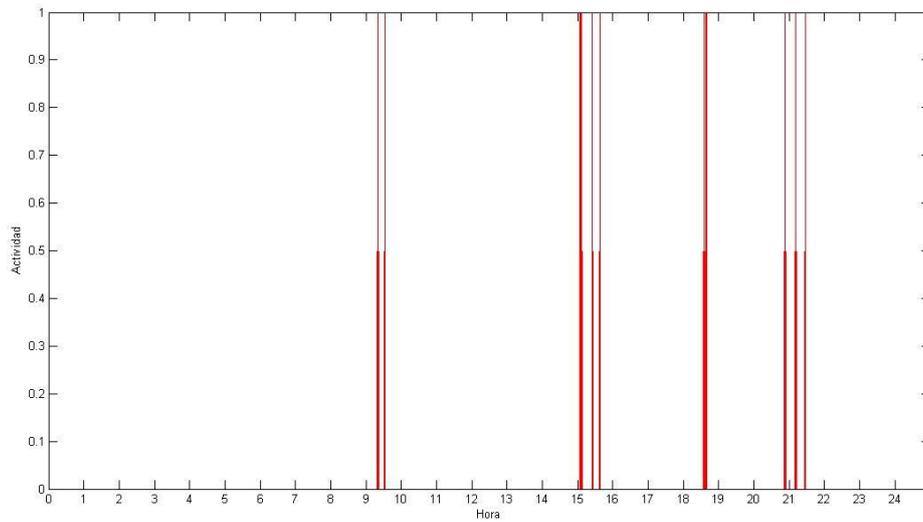


Gráfico 10. Registro de las comidas del segundo día con el Plogg de la nevera

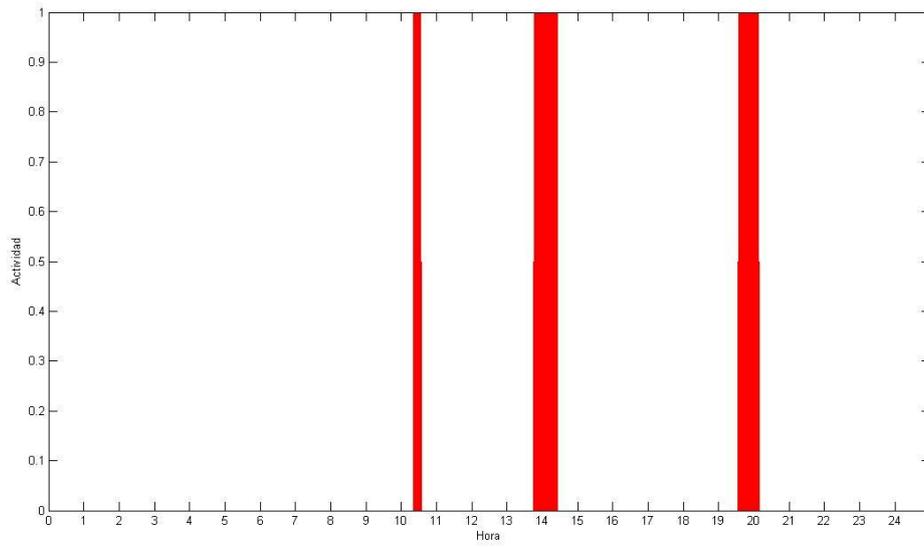


Gráfico 11. Registro de las comidas del tercer día con el Plogg de la radio

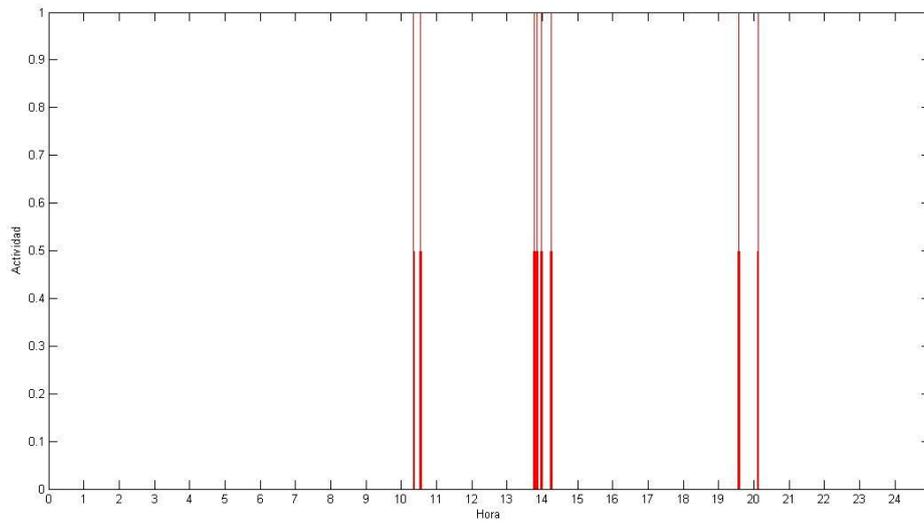


Gráfico 12. Registro de las comidas del tercer día con el Plogg de la nevera

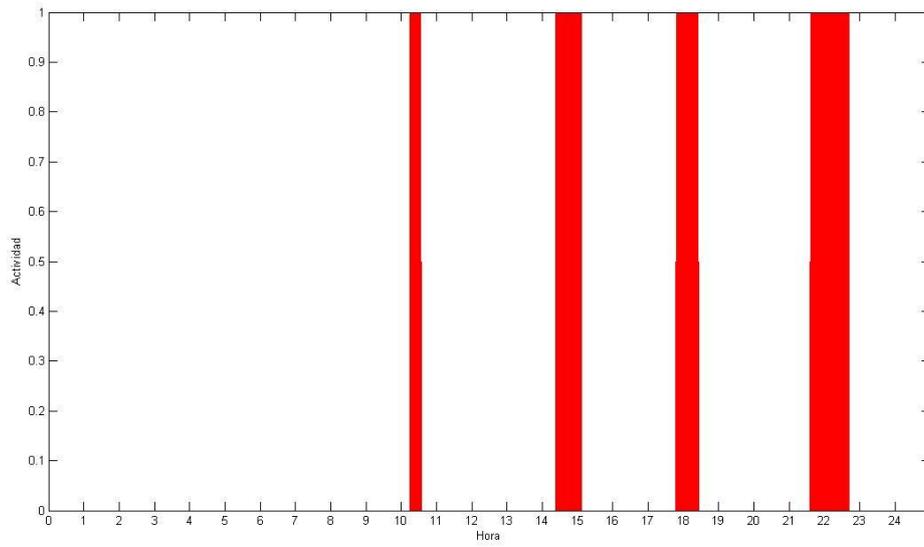


Gráfico 13. Registro de las comidas del cuarto día con el Plogg de la radio

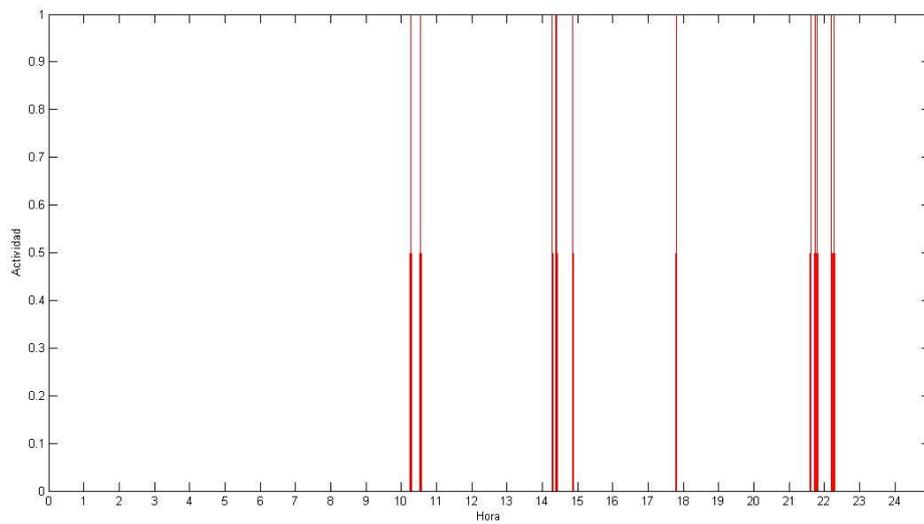


Gráfico 14. Registro de las comidas del cuarto día con el Plogg de la nevera

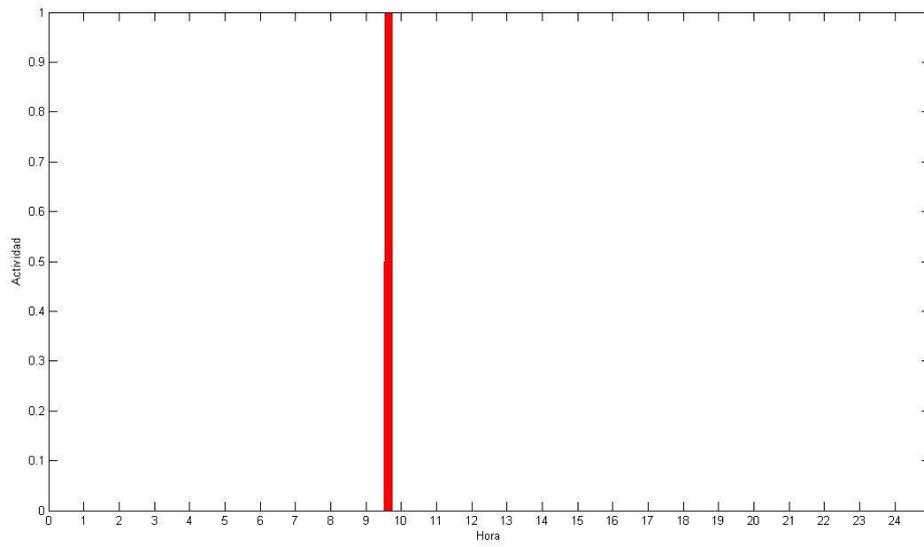


Gráfico 15. Registro de las comidas del quinto día con el Plogg de la radio

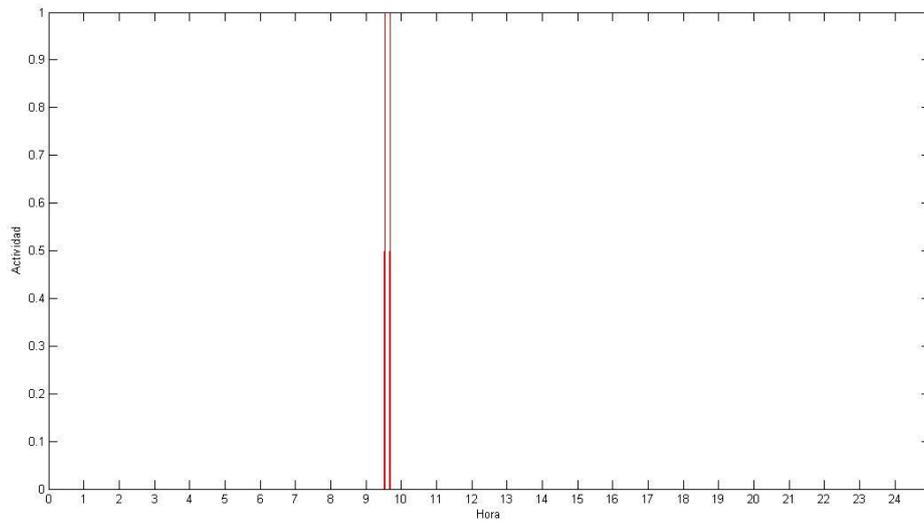


Gráfico 16. Registro de las comidas del quinto día con el Plogg de la nevera

4.5.3. Ocio:

En lo referente a las actividades relacionadas con el ocio, se estudió el tiempo que la persona dedicó a ver la televisión siendo un total de 19 horas y 22 minutos durante el periodo de pruebas, lo que supone un 22,13% del total del tiempo. Las franjas horarias en las que se vio la televisión son las siguientes (Gráficos 17, 18, 19, 20, 21).

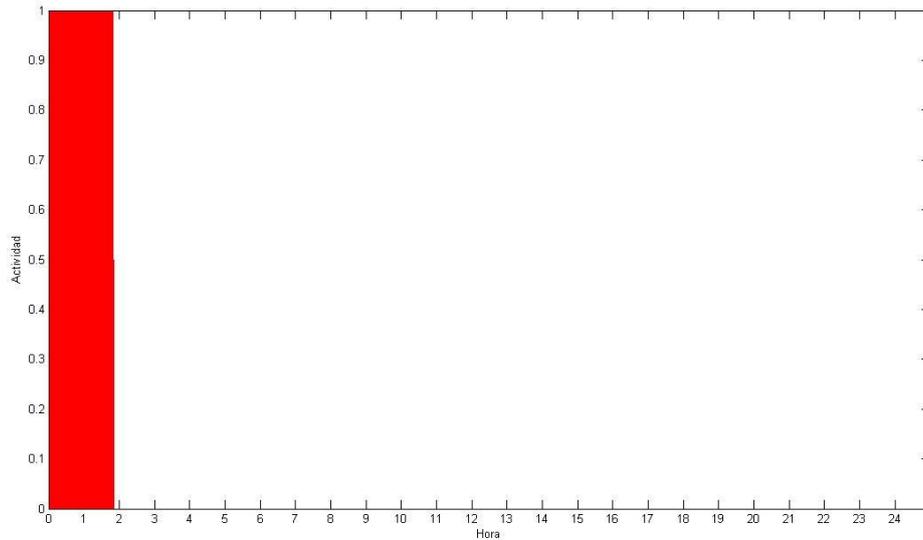


Gráfico 17. Tiempo que se dedica a ver la televisión el primer día

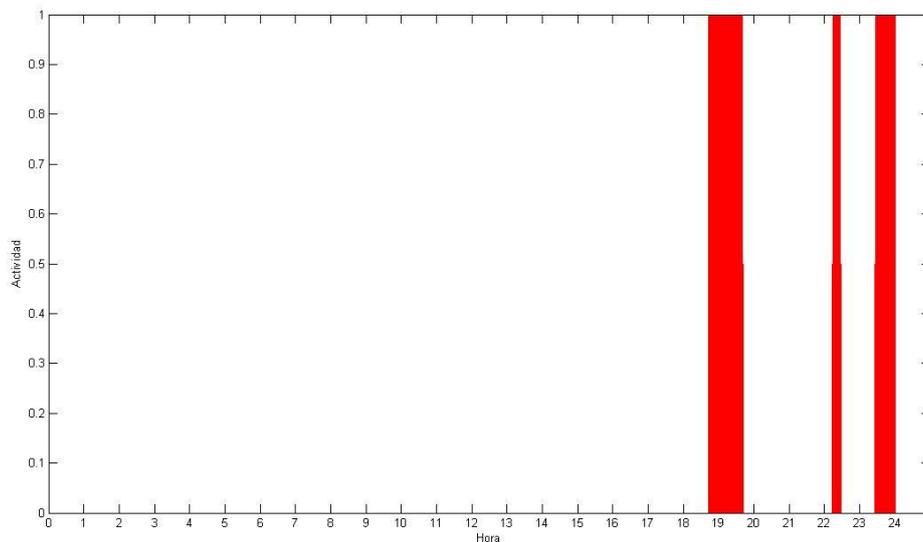


Gráfico 18. Tiempo que se dedica a ver la televisión el segundo día

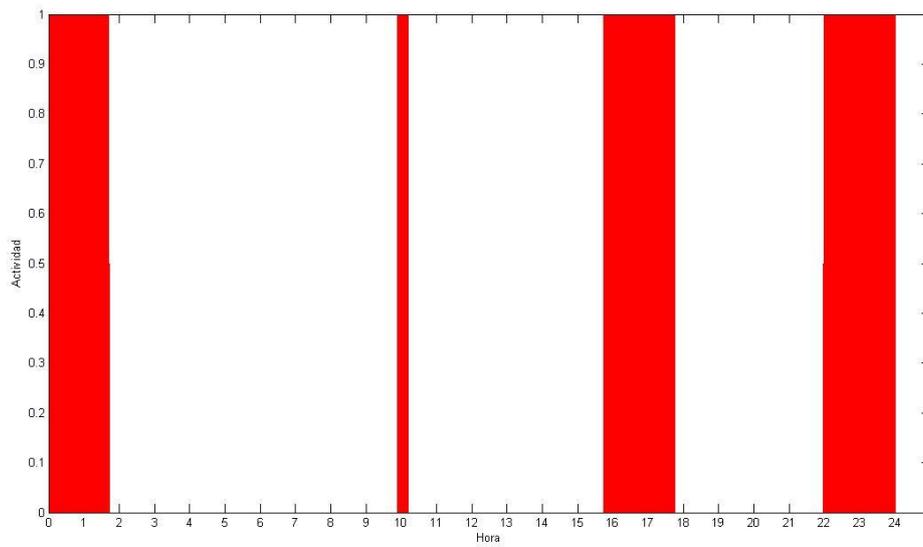


Gráfico 19. Tiempo que se dedica a ver la televisión el tercer día

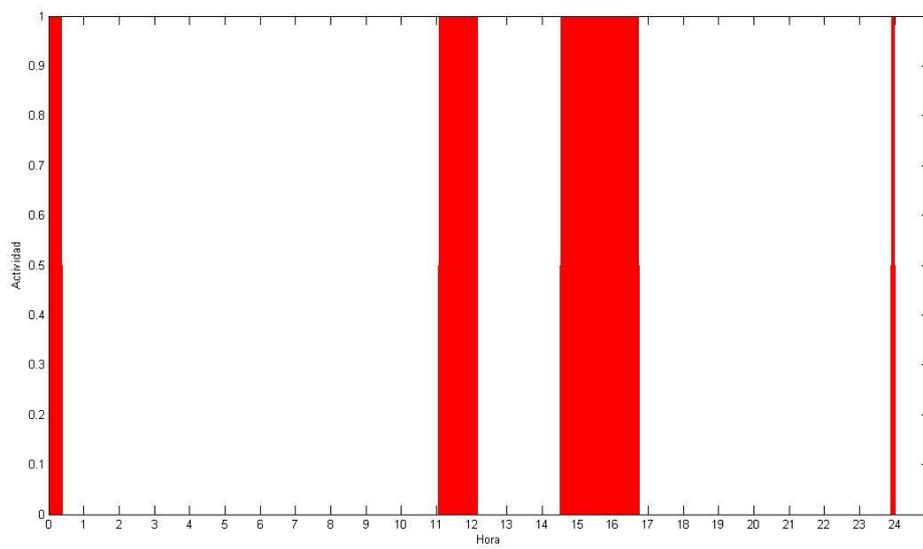


Gráfico 20. Tiempo que se dedica a ver la televisión el cuarto día

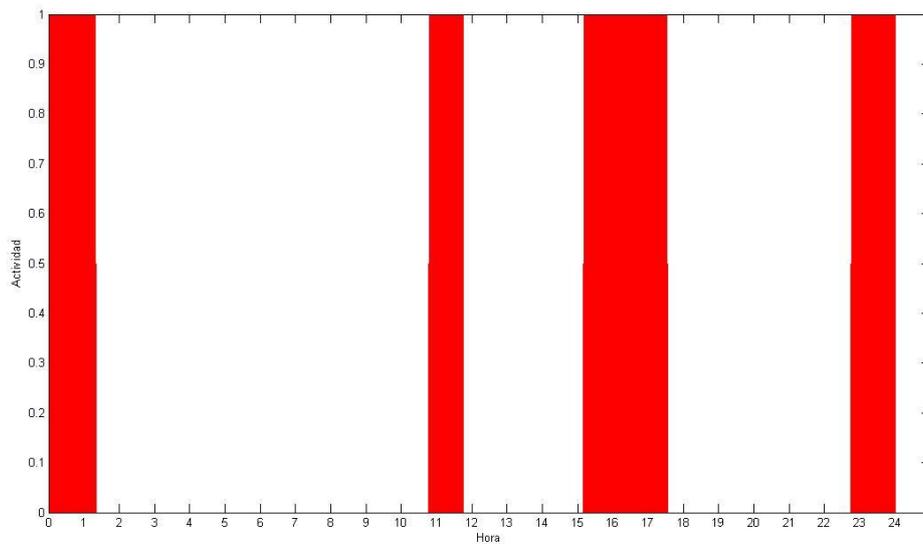


Gráfico 21. Tiempo que se dedica a ver la televisión el quinto día

En estos gráficos se muestra que el individuo solía ver la televisión en franjas horarias parecidas todos los días. Para observar mejor este hecho se representan todos los días en un mismo gráfico, cada día con un color distinto. El factor de coincidencia calculado de todos los días que vio la televisión es del 78,54% (Gráfico 22).

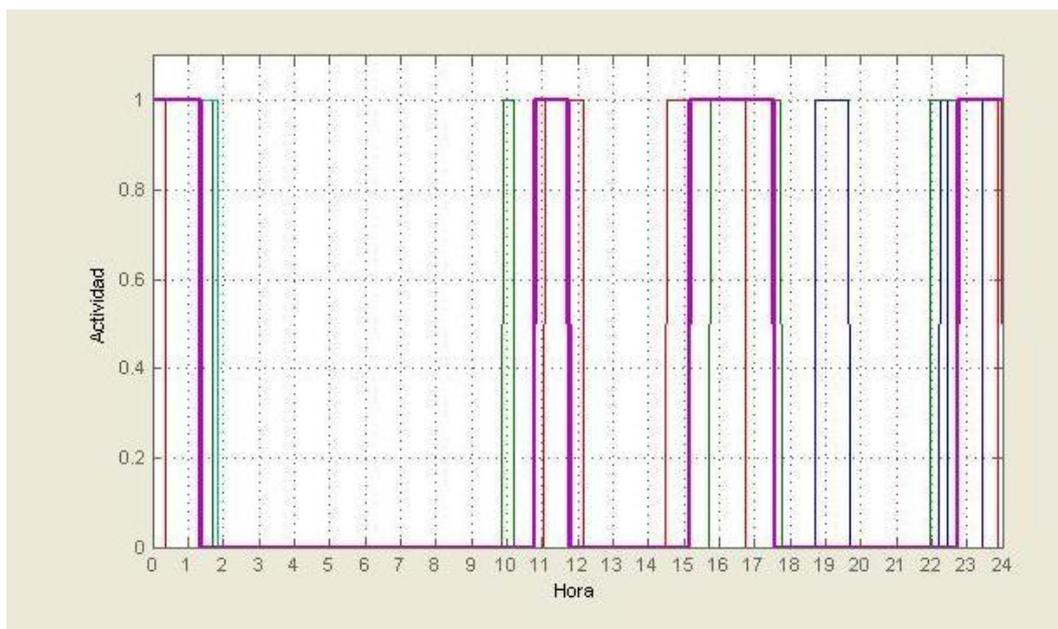


Gráfico 22. Tiempo que dedica a ver la televisión a lo largo del estudio

4.5.4. Entrada/Salida de casa:

Con el sensor de posición del vestíbulo se obtenía información de los periodos en los que el usuario entraba o salía del hogar. Estos sensores, la primera vez que se activaban correspondían con la salida del usuario, mientras que la segunda indicaba el regreso al hogar (Gráficos 23, 24). También se podían interpretar estos datos como el tiempo que estaba el usuario fuera de la casa (Gráficos 25, 26).

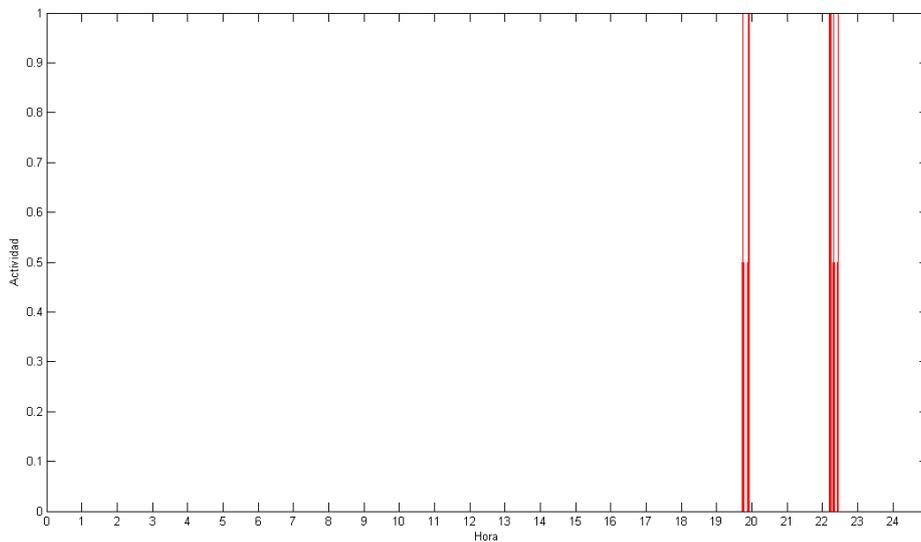


Gráfico 23. Registro de las entradas y salidas del hogar del primer día

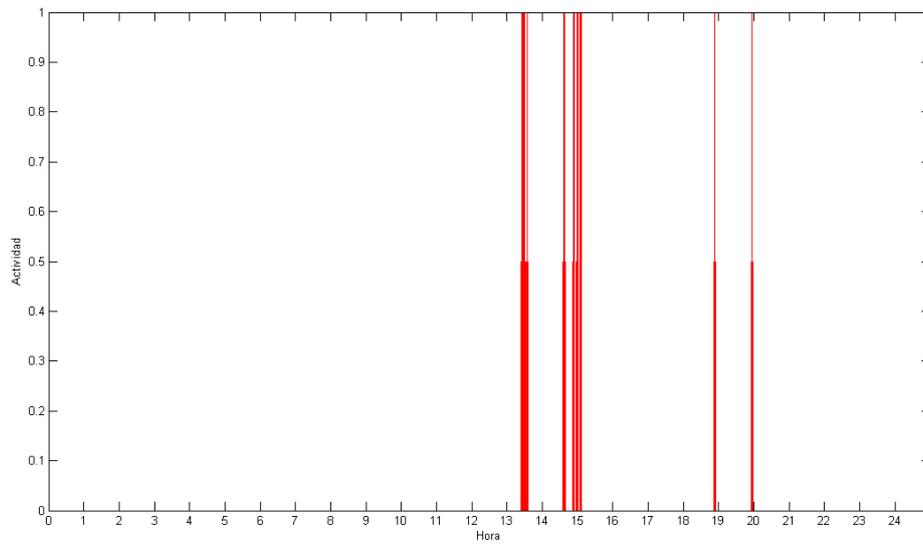


Gráfico 24. Registro de las entradas y salidas del hogar del segundo día

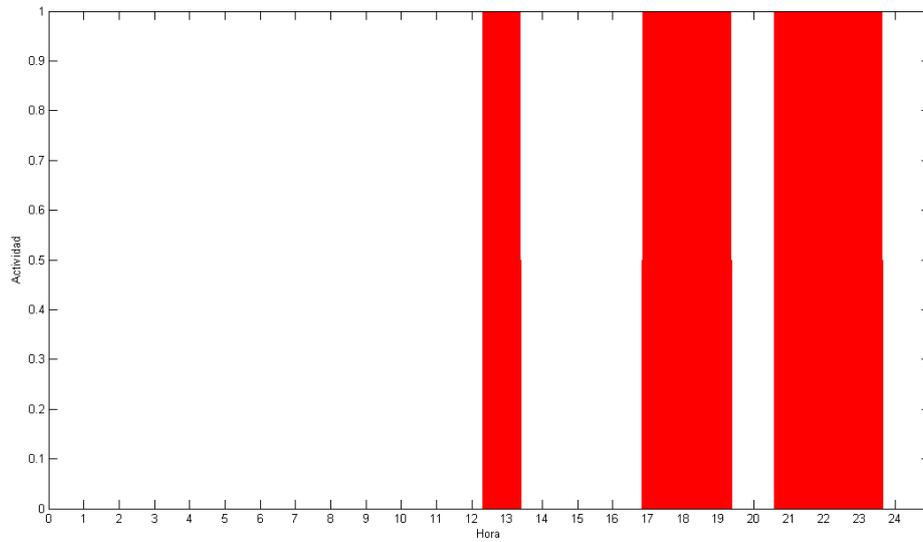


Gráfico 25. Tiempo que ha estado fuera del hogar el tercer día

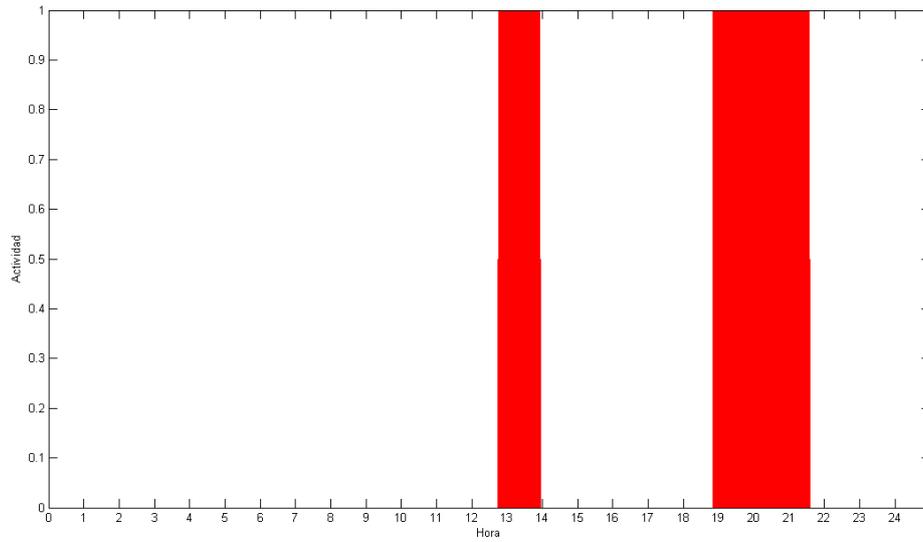


Gráfico 26. Tiempo que ha estado fuera del hogar el cuarto día

4.5.5. Ir al baño:

Con el sensor de posición del baño, se sabía que el usuario estaba en él, pero no se podía diferenciar qué actividad estaba realizando (Gráficos 27, 28, 29, 30, 31). Por lo que la solución, hubiese sido poner diversos sensores: uno en el lavabo, otro en el váter y un último en la ducha.

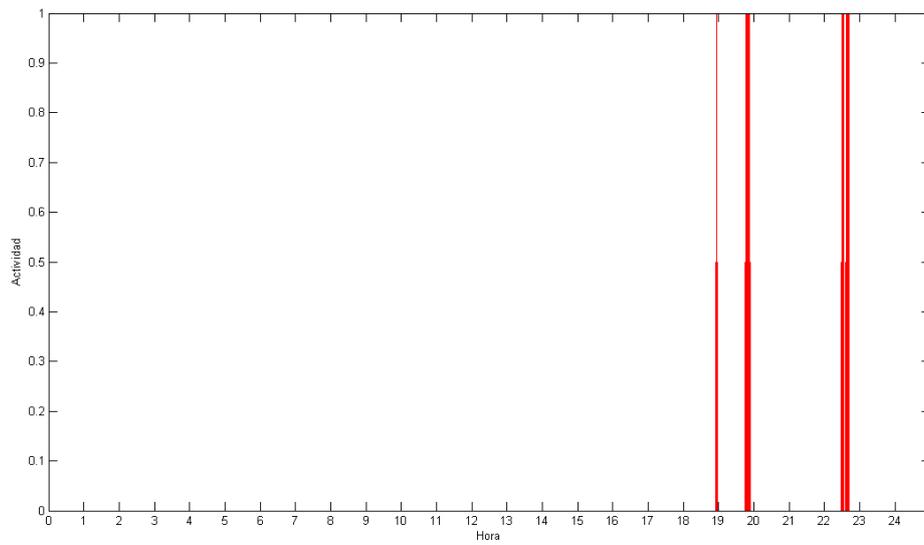


Gráfico 27. Uso del baño el primer día

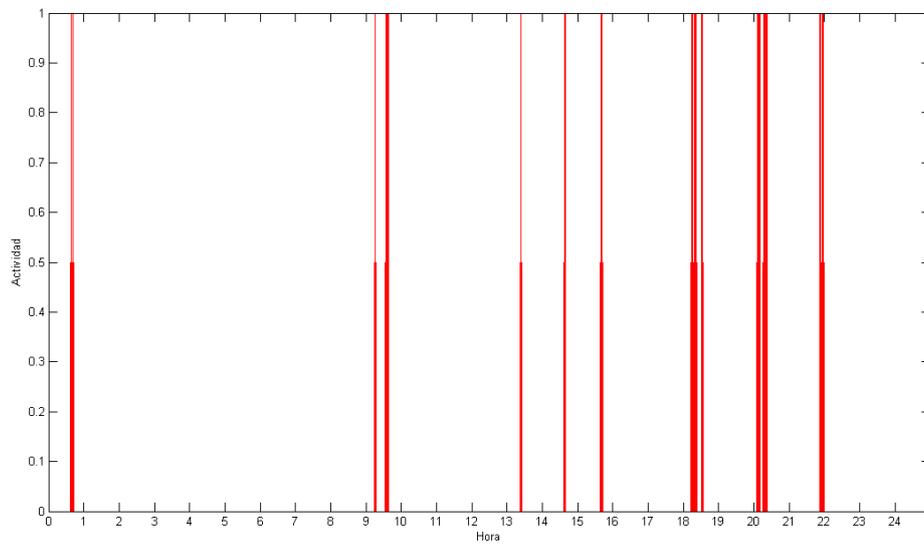


Gráfico 28. Uso del baño el segundo día

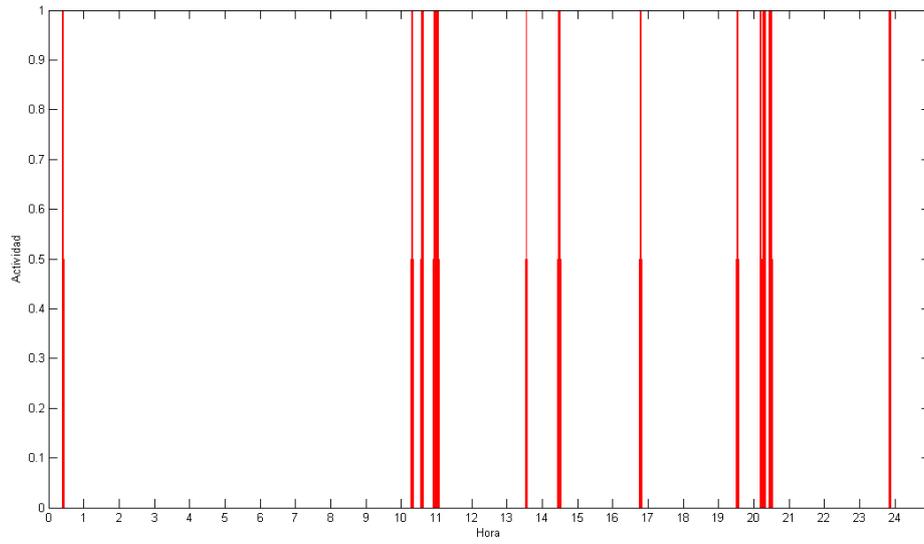


Gráfico 29. Uso del baño el tercer día

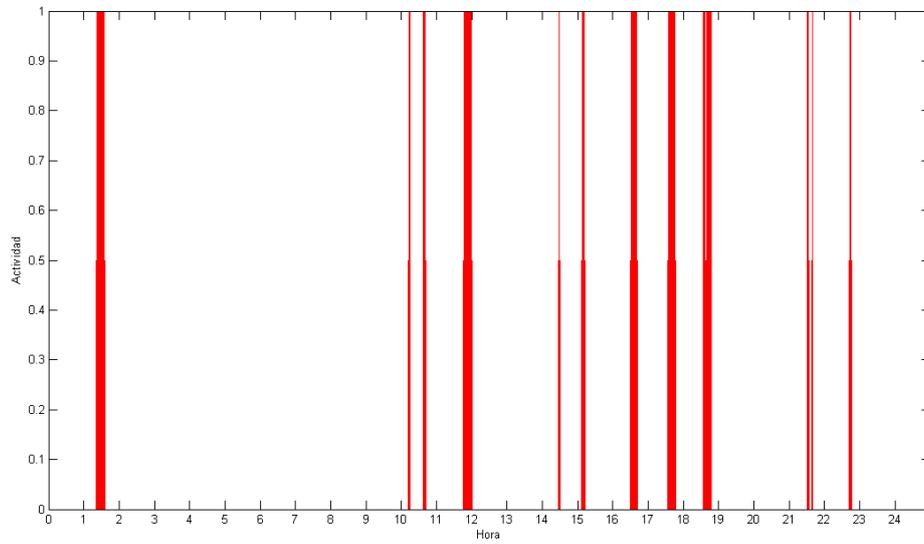


Gráfico 30. Uso del baño el cuarto día

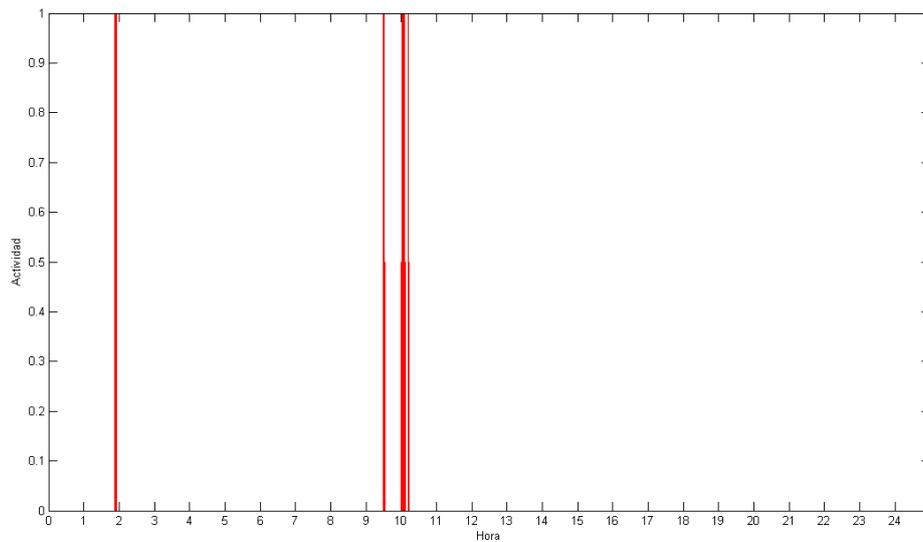


Gráfico 31. Uso del baño el quinto día

4.5.6. Descanso:

Para detectar el descanso se tenía una lamparita de noche que el usuario encendía y apagaba antes de que se fuera a dormir y en cuanto se levantaba. Mediante este parámetro se podía saber cuándo se acostaba y cuando se levantaba. Por la diferencia se podía calcular el tiempo que descansaba. A continuación se muestran los periodos de sueño del usuario, como el experimento duro cinco días le corresponden cuatro noches (Gráficos 32, 33, 34, 35).

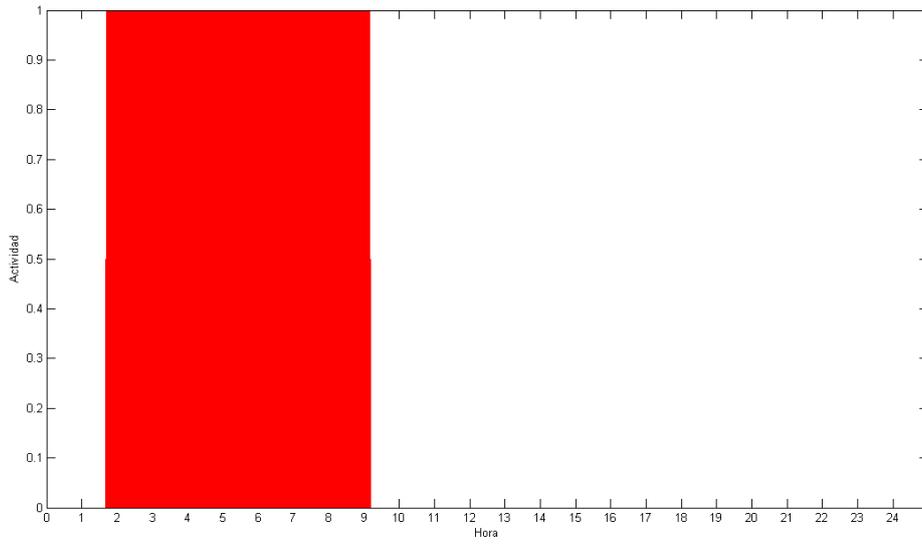


Gráfico 32. Descanso primera noche

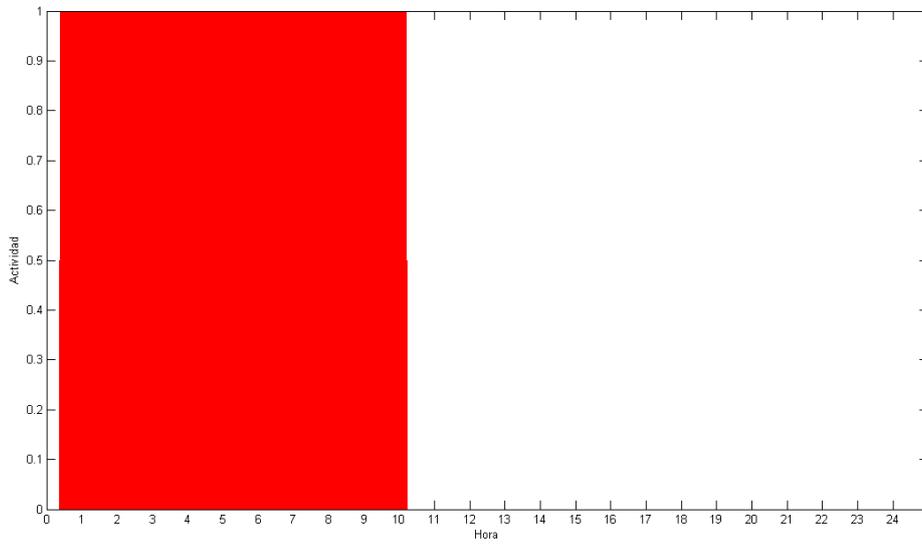


Gráfico 33. Descanso segunda noche

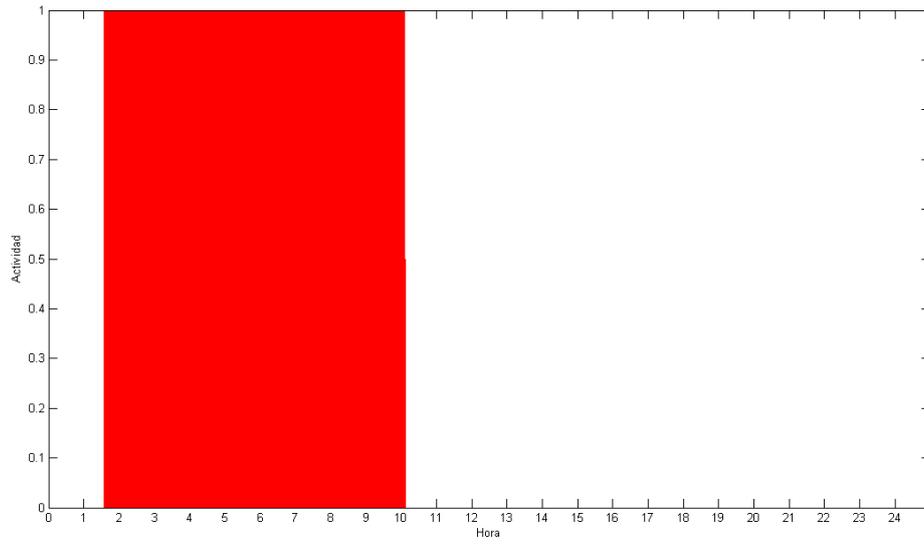


Gráfico 34. Descanso tercera noche

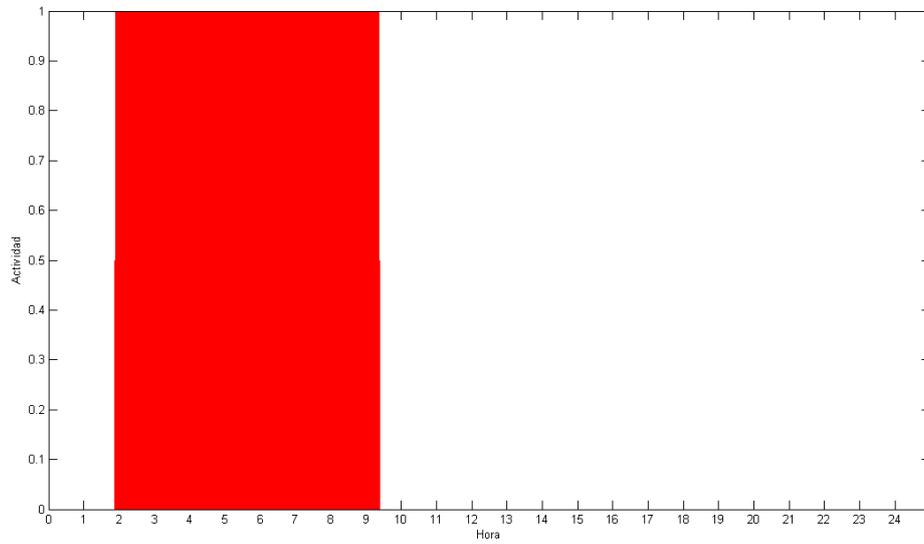


Gráfico 35. Descanso cuarta noche

4.5.7. Un día completo:

Una forma de representar los datos que se han descrito de forma detallada anteriormente, sería en una única gráfica con las distintas actividades desarrolladas por el usuario a lo largo de un día. A continuación se muestran las principales actividades que lleva a cabo el usuario el cuarto día (Gráfico 36).

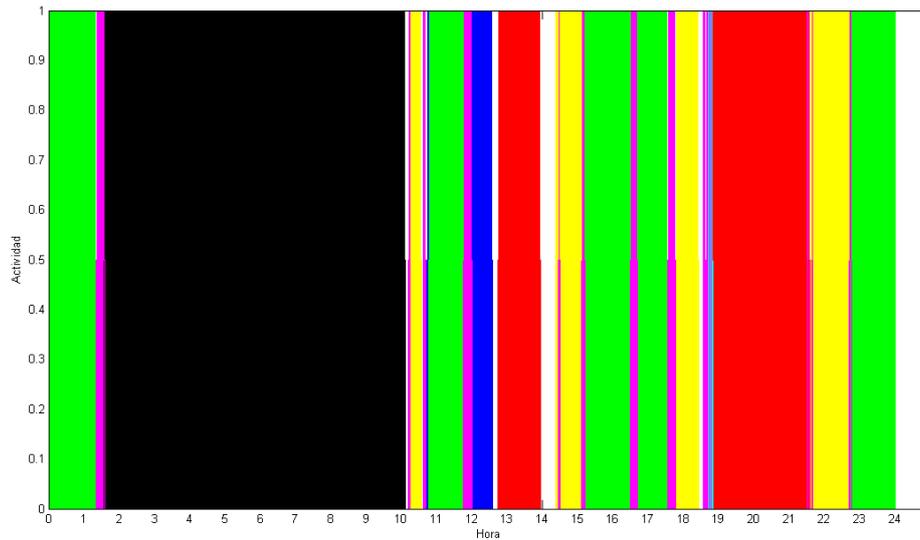


Gráfico 36. Actividades del cuarto día

- Color negro: descanso.
- Color amarillo: alimentación.
- Color azul: pone la lavadora.
- Color verde: ve la televisión.
- Color rosa: va al baño.
- Color rojo: está fuera del hogar.
- Color azul claro: se ducha.

CAPÍTULO 5:

Conclusiones

5.1 Conclusiones del trabajo:

La herramienta que se ha desarrollado en este proyecto puede ser útil para los servicios sociales por varias razones. La primera, porque se usan unos sensores sencillos que no requieren obras en su instalación y pueden colocarse en cualquier casa independiente de su distribución interna. Además, se ajustan fácilmente a cada hogar mediante un breve periodo de prueba en el que el usuario realiza unas acciones para establecer los márgenes de detección de actividad, como pueden ser los distintos consumos para un mismo electrodoméstico dependiendo del hogar. Y por último, la representación de los datos se puede hacer de la forma que más interese al agente social, por ejemplo, se puede detectar que la persona entra y sale de casa (los momentos concretos) o mostrar los periodos presencia/ausencia de la persona en la casa.

La herramienta se ha diseñado con la idea de ser instalada en hogares de personas mayores que normalmente, suelen tener unos horarios fijos para hacer las distintas actividades cotidianas. Aunque esto puede depender de cada caso concreto, suele ser una buena forma de valorar que la persona se encuentra bien de salud. Para ello, se puede configurar la aplicación de tal manera que sólo se muestren las desviaciones importantes de los hábitos cotidianos. Ésto se hace fijando unos márgenes de tiempo para las comidas o para dormir y el programa muestra sólo si se ha salido de ellos, ya que podría reflejar algún trastorno alimentario o del sueño que representaría un problema de salud. También se pueden analizar los datos como han hecho en el artículo “An algorithm for the automatic detection of health conditions [21]” en el que con los

patrones de horas que la persona ha visto la televisión se deduce si está sana o tiene algún problema. Otra opción sería observar la frecuencia con la que acude al baño y si varía mucho de repente puede indicar que tiene algún problema de salud.

Durante el análisis de los datos se ha observado alguno muy significativo como puede ser detectar que se ducha por el aumento de temperatura que se produce en el baño.

Las pruebas se han realizado durante el mes de octubre por lo que el tiempo no era ni muy caluroso ni muy frío. Aunque no se han mostrado los resultados del sensor de temperatura ubicado en el vestíbulo sería útil para comprobar si la persona enciende la calefacción o pasa frío en invierno, o en caso de que tenga aire acondicionado, si lo enciende o pasa calor durante el verano.

5.2 Cumplimiento de objetivos:

Tras la finalización de este proyecto se han cumplido los objetivos principales: haber realizado una revisión bibliográfica con las soluciones que dan otros autores al problema de detectar actividades y haber desarrollado un sistema prototipo funcionando y probado. Este sistema consta de dos partes, una de recogida de datos de los Ploggs que funciona de manera continua y otra, de unos Scripts que nos permiten traducir los datos recogidos por los Ploggs o los multisensores a actividades desarrolladas por el usuario.

Se ha logrado superar los objetivos parciales planteados:

- Se ha realizado un completo estudio del estado del arte, obteniendo una gran variedad de soluciones para el problema inicial propuesto.
- Se ha conseguido un manejo adecuado de las herramientas necesarias para realizar el proyecto como puede ser Matlab o Telegesis Terminal.
- Se ha aprendido el funcionamiento del protocolo de comunicación Zigbee, así como a crear y configurar redes.
- Se ha comprendido el manejo de los Plogg.

- Se ha hecho un estudio de ubicación de los sensores para conseguir una mejor detección de la actividad.
- Se ha desarrollado un software para la recogida de datos de los Plogg de manera continua, con una gran robustez.
- Se ha instalado en un piso piloto el sistema para comprobar su correcto funcionamiento, así como para realizar una recogida de datos de modo continuo.
- Se ha diseñado un software que analiza los datos obtenidos y diferencia actividades.
- Se han analizado los datos recogidos obteniendo conclusiones muy interesantes.

Por todo lo comentado, se puede concluir que se han cumplido de manera satisfactoria todos los objetivos planteados al inicio del proyecto.

5.3 Desarrollos futuros:

En un futuro, este proyecto se puede mejorar fundamentalmente en dos apartados: la recogida de los datos y su posterior tratamiento.

En lo referente a la parte de la recogida de datos, las mejoras podrían realizarse de las siguientes formas:

- Colocando un sensor de presencia en cada habitación para saber dónde está el individuo. Además añadiría uno en el váter, otro en el lavabo y un último en la ducha, ya que es muy interesante poder diferenciar esas tres actividades que se realizan en el baño.
- Añadiendo más Ploggs. Tener uno móvil que el usuario conectara donde quisiera, por ejemplo, si conectáramos en él el aspirador tendríamos información sobre actividades relacionadas con la limpieza.

- Poniendo contactores magnéticos en las puertas de los armarios y cajones de la cocina para saber que usa los utensilios que hay en ellos o saca la comida. Y en los armarios y cajones de su habitación para poder saber si se viste.
- Modificando el código del programa de recogida de datos, pidiendo un dato a un Plogg y sin esperar a recibirlo, pedirlo a otro Plogg. Aparte habría que administrar la cola de datos recibidos. De esta manera, se conseguiría que trabajase de forma paralela, en lugar de secuencial. Así, se aumentaría la cantidad de datos recogidos al minuto por los Plogg.
- Trasladando el programa de recogida de datos de Matlab a Java, para que pudiese funcionar en cualquier ordenador, sin la necesidad de tener instalado Matlab.
- Recogiendo datos de forma continua durante un largo periodo de tiempo y a varios usuarios.

En el apartado del tratamiento de los datos, las mejoras irían encaminadas a:

- Obtener muchos datos para hacer estudios estadísticos representativos de patrones de conducta del usuario.
- Procesar los datos en una red neuronal para realizar un análisis de reconocimiento de patrones y poder analizar posibles desajustes en los horarios del usuario.
- Para evitar falsos positivos en las salidas del hogar sería importante que, además de comprobar que se activa el sensor de posición del vestíbulo, se observara que no hubiese actividades dentro de la casa, que sería lo normal debido a la ausencia del individuo.

Sería muy interesante que los datos que recogen los sensores en el hogar fueran enviados a través de Internet o de un módulo de telefonía móvil, una vez al día, al centro de tratamiento de los datos, para así poder seguir en tiempo más o menos real las evoluciones del usuario y detectar posibles problemas que pueda tener.

5.4 Conclusiones personales:

La realización de este proyecto fin de carrera me ha supuesto un gran aprendizaje, tanto de herramientas de desarrollo como de métodos de trabajo.

He estudiado y alcanzado un cierto nivel en el manejo de herramientas como Matlab, el terminal de Telegesis, así como de las comunicaciones Zigbee o el conocimiento de los Plogg.

En resumen he conseguido tener una visión global del proceso de investigación y desarrollo de prototipos con el fin de encontrar nuevas soluciones a problemas planteados.

Además, he llevado a cabo la redacción de una memoria, que es una parte muy importante en cualquier trabajo de investigación. Ya que hay que conseguir desarrollar programas o generar datos pero también hay que saber interpretar y transmitir esos resultados.

Con todo esto, creo que mi formación académica de la carrera se ha visto completada al realizar y defender este proyecto.

BIBLIOGRAFÍA

- [1] Tecnodiscap.
<http://tecnodiscap.unizar.es>
- [2] Instituto de Investigación en Ingeniería de Aragón.
<http://i3a.unizar.es>
- [3] S. Katz, A. B. Ford, R. W. Moskowitz, B.A. Jackson and M.W. Jaffe. “Studies of illness in the aged: The index of ADL, a standardized measure of biological and psychosocial function”. *Journal of the American Medical Association*, pp. 914-919, 1963.
- [4] S. Katz, T. D. Downs, H. R. Cash and R. C. Grotz. “Progress in development of the index of ADL”. *The Gerontologist*, pp. I, 20-30, 1970.
- [5] S. Katz. “Assessing self-maintenance: Activities of daily living, Mobility, and Instrumental activities of daily living”. *Journal of the American Geriatrics Society*, vol. 31, nº 12, pp. 721-727, 1983.
- [6] Plogg wireless energy management.
<http://www.plogg.co.uk/>
- [7] Plogg guía de instalación.
<http://www.plogg.co.uk/docs/EO-ZIGBEE-SETUP%20v2011.pdf>
- [8] Plogg especificaciones técnicas.
<http://www.plogg.co.uk/docs/TechnicalSpecificationV02.pdf>
- [9] Telegesis.
<http://www.telegesis.com/>
- [10] Matlab.
<http://www.mathworks.es/>

- [11] Zigbee Alliance.
<http://www.zigbee.org/>
- [12] Google académico.
<http://scholar.google.es/>
- [13] ISI Web of Knowledge.
<http://www.accesowok.fecyt.es/>
- [14] IEEE Xplore.
<http://ieeexplore.ieee.org/Xplore/guesthome.jsp?reload=true>
- [15] X. H. Binh Le, M. Di Mascolo, A. Gouin and N. Noury. "Health smart home for elders – A tool for automatic recognition of activities of daily living". *30th IEEE-EMBS, Vancouver*, pp. 3316-3319, 2008.
- [16] M. Chan, C. Hariton, P. Ringiard and E. Campo. "Smart house automation system for the elderly and the disabled". *IEEE 0-7803-2559-1/95*, pp. 1586-1589, 1995.
- [17] T. S. Barger, D. E. Brown and M. Alwan. "Health-status monitoring through analysis of behavioral patterns". *IEEE Transactions on systems, man, and cybernetics-part A: systems and humans*, vol. 35, n° 1, pp. 22-27, 2005.
- [18] N. Zouba, F. Bremond and M. Thonnat. "Multisensor fusion monitoring elderly activities at home". *IEEE 978-0-7695-3718-4/09*, pp. 98-103, 2009.
- [19] A. Fleury, M. Vacher and N. Noury. "SVM-based multimodal classification of activities of daily living in health smart homes: sensors, algorithms, and first experimental results". *IEEE transactions on information technology in biomedicine*, vol. 14 n° 2, pp. 274-283, 2010.
- [20] N. Zouba, F. Bremond and M. Thonnat. "An activity monitoring system for real elderly at home: Validation study". *IEEE 978-0-7695-4264-5/10*, pp. 278-285, 2010.
- [21] M. Nambu, K. Nakajima, M. Noshiro and T. Tamura. "An algorithm for the automatic detection of health conditions". *IEEE engineering in medicine and biology magazine*, pp. 38-42, 2005.
- [22] M. Ogawa, R. Suzuki, S. Otake, T. Izutsu, T. Iwaya and T. Togawa. "Long-term remote behavioral monitoring of the elderly using sensors installed in domestic houses". *IEEE-EMBS/BMES2002, Houston*, pp. 1853-1854, 2002.

- [23] M. C. Casado, “Manual básico de Matlab”, Servicios informáticos U.C.M. apoyo a investigación y docencia.
- [24] Grupo FMI. “Un curso de Matlab”. Departamento de matemática aplicada, Universidad de Zaragoza.
- [25] J. García, J. I. Rodríguez y J. Vidal. “Aprenda Matlab 7.0 como si estuviera en primero”. Escuela Técnica Superior de Ingenieros Industriales, Universidad Politécnica de Madrid.
- [26] “Que es Matlab”. Universidad Distrital Francisco José de Caldas.
- [27] Documentación de Matlab.
http://www.mathworks.es/help/?s_cid=global_nav
- [28] Tutoriales de Matlab.
http://www.mathworks.es/academia/student_center/tutorials/?s_cid=global_nav
- [29] Foro de preguntas y respuestas sobre Matlab
http://www.mathworks.es/matlabcentral/answers/?s_cid=global_nav
- [30] Panasonic MP Motion Sensor.
<http://es.mouser.com/panasonicnapiion/>
- [31] Datasheet LM75.
<http://www.datasheetcatalog.org/datasheet/nationalsemiconductor/DS012658.PD>
- [32] R. J. Casas. “Apuntes sobre Zigbee”. Departamento de Ingeniería Electrónica y Comunicaciones, Universidad de Zaragoza.
- [33] D. Domínguez. “Anexos, Control vía Zigbee de dispositivos de iluminación con protocolo DMX para salas de estimulación multisensorial”.
- [34] IEEE 802.15 Web Site.
<http://www.ieee802.org/15/pub/TG4.html>
- [35] Protocolo Zigbee.
<http://rua.ua.es/dspace/bitstream/10045/1109/1/InformeTecZB.pdf>

ÍNDICE DE FIGURAS

Figura 1. Distribución temporal del proyecto	- 15 -
Figura 2. Ritmo de la vida diaria del ocupante durante 31 días	- 17 -
Figura 3. Disposición de los sensores en la casa piloto.....	- 19 -
Figura 4. Colocación de los sensores en la casa.....	- 20 -
Figura 5. Imágenes del tiempo que vio la televisión	- 24 -
Figura 6. Diagrama del sistema.....	- 27 -
Figura 7. Diagrama del proceso de desarrollo de la herramienta	- 28 -
Figura 8. Interface de Matlab	- 30 -
Figura 9. Terminal de Telegesis.....	- 33 -
Figura 10. Plogg	- 37 -
Figura 11. Multisensor	- 39 -
Figura 12. Estructura funcional de Zigbee sobre IEEE 802.15.4.....	- 41 -
Figura 13. Topologías de red.....	- 44 -
Figura 14. Reparto de direcciones en una red	- 45 -
Figura 15. Perfiles Zigbee	- 47 -
Figura 16. Actividades y sensores.....	- 49 -
Figura 17. Distribución de los sensores en el piso	- 51 -
Figura 18. Recogida de datos de los multisensores.....	- 53 -
Figura 19. Proceso de recogida de datos de los Ploggs.....	- 55 -
Figura 20. Fichero de texto con los datos registrados de un Plogg	- 56 -
Figura 21. Estructura de las direcciones MAC.....	- 56 -
Figura 22. Función de abrir el puerto COM.....	- 57 -

Figura 23. Control de si el dato falla durante la comunicación Zigbee.....- 58 -

Figura 24. Control de fallos en el puerto serie- 59 -

Figura 25. Función mydispcallback que se activa con la llegada de datos al puerto serie.....- 60 -

Figura 26. Diagrama de flujo principal- 61 -

Figura 27. Diagrama de flujo de la interrupción- 62 -

Figura 28. Vector con el dato del multisensor.....- 64 -

Figura 29. Diagrama Script de tratamiento de datos de los multisensores.....- 65 -

Figura 30. Diagrama global del script de tratamiento de datos de Ploggs- 66 -

Figura 31. Diagrama de cómo se guarda el dato de un Plogg- 67 -

ÍNDICE DE GRÁFICOS

Gráfico 1. Ducha del primer día.....	- 73 -
Gráfico 2. Ducha del segundo día.....	- 73 -
Gráfico 3. Ducha del tercer día.....	- 74 -
Gráfico 4. Ducha del cuarto día.....	- 74 -
Gráfico 5. Registro de los días que no se ha puesto la lavadora.....	- 75 -
Gráfico 6. Registro del cuarto día que fue en el que se utilizó la lavadora.....	- 75 -
Gráfico 7. Registro de las comidas del primer día con el Plogg de la radio.....	- 77 -
Gráfico 8. Registro de las comidas del primer día con el Plogg de la nevera.....	- 77 -
Gráfico 9. Registro de las comidas del segundo día con el Plogg de la radio.....	- 78 -
Gráfico 10. Registro de las comidas del segundo día con el Plogg de la nevera.....	- 78 -
Gráfico 11. Registro de las comidas del tercer día con el Plogg de la radio.....	- 79 -
Gráfico 12. Registro de las comidas del tercer día con el Plogg de la nevera.....	- 79 -
Gráfico 13. Registro de las comidas del cuarto día con el Plogg de la radio.....	- 80 -
Gráfico 14. Registro de las comidas del cuarto día con el Plogg de la nevera.....	- 80 -
Gráfico 15. Registro de las comidas del quinto día con el Plogg de la radio.....	- 81 -
Gráfico 16. Registro de las comidas del quinto día con el Plogg de la nevera.....	- 81 -
Gráfico 17. Tiempo que se dedica a ver la televisión el primer día.....	- 82 -
Gráfico 18. Tiempo que se dedica a ver la televisión el segundo día.....	- 82 -
Gráfico 19. Tiempo que se dedica a ver la televisión el tercer día.....	- 83 -
Gráfico 20. Tiempo que se dedica a ver la televisión el cuarto día.....	- 83 -
Gráfico 21. Tiempo que se dedica a ver la televisión el quinto día.....	- 84 -
Gráfico 22. Tiempo que dedica a ver la televisión a lo largo del estudio.....	- 84 -

Gráfico 23. Registro de las entradas y salidas del hogar del primer día	- 85 -
Gráfico 24. Registro de las entradas y salidas del hogar del segundo día.....	- 86 -
Gráfico 25. Tiempo que ha estado fuera del hogar el tercer día.....	- 86 -
Gráfico 26. Tiempo que ha estado fuera del hogar el cuarto día.....	- 87 -
Gráfico 27. Uso del baño el primer día	- 88 -
Gráfico 28. Uso del baño el segundo día.....	- 88 -
Gráfico 29. Uso del baño el tercer día.....	- 89 -
Gráfico 30. Uso del baño el cuarto día.....	- 89 -
Gráfico 31. Uso del baño el quinto día.....	- 90 -
Gráfico 32. Descanso primera noche	- 91 -
Gráfico 33. Descanso segunda noche.....	- 91 -
Gráfico 34. Descanso tercera noche	- 92 -
Gráfico 35. Descanso cuarta noche	- 92 -
Gráfico 36. Actividades del cuarto día.....	- 93 -

ÍNDICE DE TABLAS

Tabla 1. Resultados del reconocimiento de actividades.....	- 21 -
Tabla 2. Reconocimiento de actividades.....	- 22 -
Tabla 3. Comandos Zigbee.....	- 35 -
Tabla 4. Actividades recogidas por el usuario	- 72 -