Intelligent Sensing Anywhere

Faculdade de Ciências e Tecnologia
da Universidade de Coimbra
Departamento de Física

AAL SAFE

Signal Processing Algorithm

Ricardo Jorge Pina Amaro
Master of Biomedical Engineering
Coimbra, September 2009

Esta tese e todo o trabalho desenvolvido ao longo do projecto, é dedicado à grande pessoa que foi o meu avô, o qual partiu no decorrer do projecto mas que jamais me abandonou.

Obrigado avô.

Abstract

Since the increase of elderly population is one of the demographic problems that entail greater concern today, at an European level a program entitled Ambient Assisted Living was established. This program aims to provide the elderly with a safer and cosier environment in their homes, thus living more autonomously and without losing their privacy.

The challenge of this project was taking advantage of a high technology present in the market, the Wii remote, and making it useful within the concept of Ambient Assisted Living.

The system developed is called *AAL* Safe and its usefulness is detecting falls monitoring daily routine activities' and calculating the energy expenditure.

The system consists of a remote of the Wii console that communicates by Bluetooth with a computer, where the data are acquired and processed.

This thesis provides a description of the processing methods, and algorithms developed to detect the fall, to identify the activities of daily routine, and compute the energy expenditure, together with its implementation mode.

Resumo

Sendo o aumento de população idosa, um dos problemas demográficos que acarreta maiores preocupações actualmente, foi criado a nível europeu um programa intitulado Ambient Assisted Living que tem como objectivo desenvolver soluções um ambiente mais seguro e acolhedor às pessoas idosas. Assim podem viver de forma mais autónoma e sem perda de privacidade.

O desafio deste projecto foi tirar partido de uma tecnologia em alta no mercado, que é o comando da consola Wii, e conseguir deste modo torna-lo útil dentro do conceito do Ambient Assisted Living.

O sistema desenvolvido intitula-se *AAL* Safe, e tem como funções detectar quedas, monitorizar as actividades da rotina diária assim como calcular o dispêndio energético.

O sistema é constituído por um comando da consola Wii que comunica via Bluetooth com um computador, onde os dados são adquiridos e processados.

Esta tese fornece uma descrição dos modos de processamento e dos algoritmos desenvolvidos para a detecção da queda, identificação das actividades da rotina diária e cálculo do dispêndio energético, assim como o seu modo de implementação.

Acknowledgments

First of all, I would like to thank my parents, my sister and my brother-in-law for all the support and incentive they have given to me during this important period of my life.

In second I would like to thank Engineer Soraia Rocha for her good leadership, which represents a great help for the project development and conclusion.

I'd like to thank all ISA Co-workers for the hospitality, and particularly to Engineer Rafael Simões for all the help that he has offered.

A big thank to Catarina Pereira, for her attention and availability that never ceased to exist.

I'd like to thank Engineer José Malaquias for the interest he has shown in our project.

Thanks to all my friends, that each in their own way, have always given me the support and help during this project. Thank you for all the hours we have shared, you all know who you are.

Finally, and perhaps my biggest thanks goes to my friend and project partner Cátia Costa, for her friendship during the hardest times, and also for her dedication and all the hard work she has put in this project.

Contents

List of Tables

List of Figures

Acronyms

AAL	Ambient Assisted Living
Wiimote	Wii Remote
TA	Tri-Axial
PC	Personal Computer
ISA	Intelligent Sensing Anywhere
CEI	Centro de Electrónica e Instrumentação
DRA	Daily Routine Activities
C#	C Sharp
AAL JP	Ambient Assisted Living Joint Program
ICT	Information and Communication Technologies
API	Application Program Interface
DOF	Degrees of Freedom
HDI	Human Interface Device
IMU	Inertial Mounted Unit
SDP	Service Discover Protocol
SMA	Signal Magnitude Area
FFT	Fast Fourier Transform
LPF	Low Pass Filter
IIR	Infinite Impulse Response
DC	Direct Current
AC	Alternate Current
GA	Gravity Acceleration
BA	Body Acceleration
SMV	Signal Magnitude Vector
FDATool	Filter Design & Analysis Tool
DeployTool	Deployment Tool
FIR	Finite Impulse Response

1 INTRODUCTION

1.1 Motivation

The Ambient Assisted Living (AAL) is a research development program that aims to improve the quality of life of the elderly people on their homes, providing a continuous support for daily life. This subject is in focus on European societies and has been inciting the development of new solutions for healthcare monitoring in the home environment.

Another reason for developing this project was the success of Wii Remote (Wiimote), due to its high technology and enumerous functionalities, which has motivated several studies and research about its capacities and applications. The most attractive Wiimote features are its ability in recognizing movements and its wireless nature.

Thus, the AAL concept associated to Wiimote technology can provide a large number of solutions to help and facilitate the daily life of elderly people. The fact that the falls represents the major cause of injuries and morbidity among the seniors motivated the students, in a first stage of the project to make the Wiimote a fall detector.

1.2 Objectives

This project aims to develop a system to acquire and transmit the data from the Tri-Axial (TA) accelerometer of the Wiimote, to a Personal Computer (PC) to be processed on real-time. The system should be divided in tree layers, the first layer incorporates the Windows Application, and the second layer incorporates the report module and processing modules which is connected to the Wiimote. The second layer communicates with the third layer of the system where the database is.

To be possible to attain this objective the project was divided in to two parts, the interface with the user and the algorithms, both of Software. A student was responsible for the algorithms developments and implementation.

In general the *AAL* Safe system, in charge of the student, had to be able to detect falls, identify Daily Routine Activities (DRA) and compute the Energy Expenditure.

1.3 Scope

This project was developed in the scope of the Master integrated in Course of Biomedical Engineering, taught on the Physics Department of the University of Coimbra. This project was realized in partnership with ISA and CEI.

1.4 Audience

This project is addressed to the supervisors and jury members, and possible future students that may continue this project.

1.5 Document Structure and Organization

This document is structured in ten chapters.

The intent of the present chapter is to introduce the reader to the project motivation and objectives.

The following chapter, Project Management, provides information about the project organization such as project members and their functions. It also provides a brief overview on the scheduling works.

The third chapter provides the reader with an overview about the problems that our solution intends to minimize, the state of art and a brief comparison between *AAL* Safe and the similar products present in the market.

On the fourth chapter the project requirements will be describe, more specifically, the Wiimote characteristics and capabilities, and its way of communication.

The fifth chapter has the purpose of describing the *AAL* Safe system architecture, specifically the physical architecture and logical architecture .

Chapter 6 – Signal Processing – In this chapter the ways of signal processing will be described.

Chapter 7 is about the algorithms and their methods of decision.

The eighth chapter intends to describe the software development both in Matlab and in C Sharp(C#).

The testes related to the algorithm performance are described in the ninth chapter.

In the last chapter the conclusion is drawn with a brief appreciation about the project and the possible future works.

2 PROJECT MANAGEMENT

2.1 Project Members

The project team was composed of two Biomedical Engineering students of the Faculty of Sciences and Technology, University of Coimbra and their supervisors.

The students were responsible for the projection and development of the *ALL* Safe system. The supervisors accompanied the project evolution and individual development of the students, being responsible for the orientation and coordination of their work.

The elements that composed the team are presented in the table.

Table 1- Project Members

Name	Contribution	Contact
Ricardo Amaro	Student	ramaro@mail.isa.pt
Cátia Costa	Student	ccosta@mail.isa.pt
Engineer Soraia Rocha	Supervisor	srocha@mail.isa.pt
Professor José Basílio Simões	Supervisor	jbasilio@mail.isa.pt
Professor Carlos Correia	Supervisor	correia@lei.fis.uc.pt
Engineer José Malaquias	Supervisor	jmalaquias@isa.pt
Engineer Rafael Simões	Supervisor	rsimoes@mail.isa.pt

2.2 Task Division

The tasks division of the project was made in the second semester. Until the project division all work was developed by the two students. When the project was divided the student Cátia Costa was responsible to develop the program software responsible to interact with the user allowing the management of the patients' data and visualise the monitoring. Ricardo Amaro was responsible to develop the algorithms and implement them in the program developed by the other student.

2.3 Project Supervising

The project was supervised by two entities, the CEI by university, and ISA, the company where the project was developed.

2.3.1 Supervising at ISA

Intelligent Sensing Anywhere (ISA), is a spinoff enterprise of the Department of Physics – University of Coimbra. As a global technology leader, the company has a Research and Development department whose main goal is to provide complete solutions in several fields such as telemetry, industrial automation, environment and healthcare. ISA was the main entity that created the project and the core of its development as it provided the human resources to assist on its progress. The integration in a company environment such as ISA was also very useful in the point of view of the student, since it could be important in future developments.

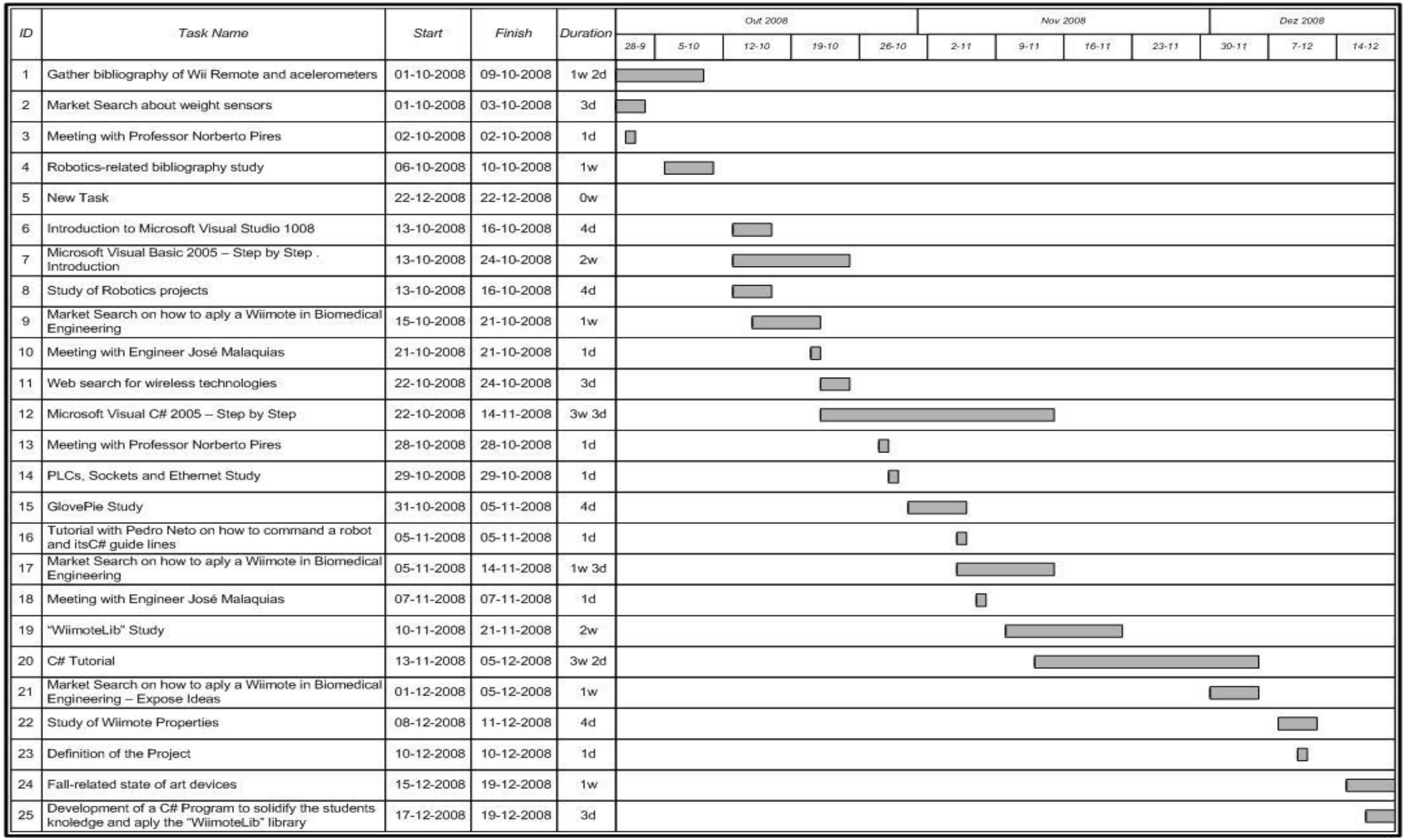
2.3.2 Supervising at CEI

Electronics Instrumentation Centre (CEI), of the University of Coimbra, is a research centre created in the Physics Department. It has a wide range of research areas, including Atomic and Nuclear Instrumentation, Biomedical Instrumentation Telemetry and Industrial Control, among others. CEI keeps close contact with a few corporations, including ISA, and acted as a bridge between the company and the University of Coimbra. The center installations operated as the student's work place, where the project was developed under the close supervision of Professor Carlos Correia.

2.4 Scheduling

The scheduling of the project activities is shown in the following tables. It is possible to see all, developed work and knowledge acquisition, achieved by the student.

Table 2- Scheduling for first semester.



ID	Task Name	Start	Finish	Duration
1	Gather bibliography of Wii Remote and acelerometers	01-10-2008	09-10-2008	1w 2d
2	Market Search about weight sensors	01-10-2008	03-10-2008	3d
3	Meeting with Professor Norberto Pires	02-10-2008	02-10-2008	1d
4	Robotics-related bibliography study	06-10-2008	10-10-2008	1w
5	New Task	22-12-2008	22-12-2008	0w
6	Introduction to Microsoft Visual Studio 1008	13-10-2008	16-10-2008	4d
7	Microsoft Visual Basic 2005 – Step by Step . Introduction	13-10-2008	24-10-2008	2w
8	Study of Robotics projects	13-10-2008	16-10-2008	4d
9	Market Search on how to aply a Wiimote in Biomedical Engineering	15-10-2008	21-10-2008	1w
10	Meeting with Engineer José Malaquias	21-10-2008	21-10-2008	1d
11	Web search for wireless technologies	22-10-2008	24-10-2008	3d
12	Microsoft Visual C# 2005 – Step by Step	22-10-2008	14-11-2008	3w 3d
13	Meeting with Professor Norberto Pires	28-10-2008	28-10-2008	1d
14	PLCs, Sockets and Ethernet Study	29-10-2008	29-10-2008	1d
15	GlovePie Study	31-10-2008	05-11-2008	4d
16	Tutorial with Pedro Neto on how to command a robot and itsC# guide lines	05-11-2008	05-11-2008	1d
17	Market Search on how to aply a Wiimote in Biomedical Engineering	05-11-2008	14-11-2008	1w 3d
18	Meeting with Engineer José Malaquias	07-11-2008	07-11-2008	1d
19	"WiimoteLib" Study	10-11-2008	21-11-2008	2w
20	C# Tutorial	13-11-2008	05-12-2008	3w 2d
21	Market Search on how to aply a Wiimote in Biomedical Engineering – Expose Ideas	01-12-2008	05-12-2008	1w
22	Study of Wiimote Properties	08-12-2008	11-12-2008	4d
23	Definition of the Project	10-12-2008	10-12-2008	1d
24	Fall-related state of art devices	15-12-2008	19-12-2008	1w
25	Development of a C# Program to solidify the students knoledge and aply the "WiimoteLib" library	17-12-2008	19-12-2008	3d

Table 3- Scheduling for second semester, part I.

ID	Task Name	Start	Finish	Duration
1	Continuation of the Development of a C# Program to program the Wiimote's features	07-01-2009	16-01-2009	3d
2	Bluetooth Protocol Study	13-01-2009	16-01-2009	4d
3	HID protocol Study	13-01-2009	16-01-2009	4d
4	Study of Articles on position detection with accelermeter	19-01-2009	23-01-2009	1w
5	1st Intercalarly Presentation Poster development and revision	27-01-2009	04-02-2009	1w 2d
6	Study of fall detectors algorithms	02-02-2009	06-02-2009	1w
7	Study of algorithms to detect DRA activities	05-02-2009	11-02-2009	1w
8	Matlab Tutorial	05-02-2009	19-02-2009	2w 1d
9	1st Intercalarly Presentation	18-02-2009	18-02-2009	1d
10	Project Division	03-03-2009	03-03-2009	1d
11	Real-Time Graph Study	02-03-2009	04-03-2009	3d
12	Re-start the study of fall detectors algorihms	09-03-2009	12-03-2009	4d
13	Meeting with Engineer Lara Osório	12-03-2009	12-03-2009	1d
14	Interruption	13-03-2009	17-03-2009	3d
15	Specification's Document	17-03-2009	27-03-2009	1w 4d
16	Search acceleration chart padrons	18-03-2009	24-03-2009	1w
17	Primitive definition of information to appear on the Windows Application	27-03-2009	01-04-2009	4d
18	Meeting with Eng. Soraia Rocha and Eng. Rafael Simões	02-04-2009	02-04-2009	1d
19	Flowchart algorithms development	03-04-2009	03-04-2009	1d
20	Fall detection Algorithm – First development	06-04-2009	15-04-2009	1w 3d
21	Presentation	15-04-2009	15-04-2009	1d
22	Meeting with Eng. Soraia Rocha and Eng. Rafael Simões	16-04-2009	16-04-2009	1d
23	Encontro Nacional de Estudantes de Engenharia Biomédica	16-04-2009	17-04-2009	2d
24	Study of processing signal lybraries to C#	20-04-2009	24-04-2009	1w
25	Matlab Studing	27-04-2009	01-05-2009	1w

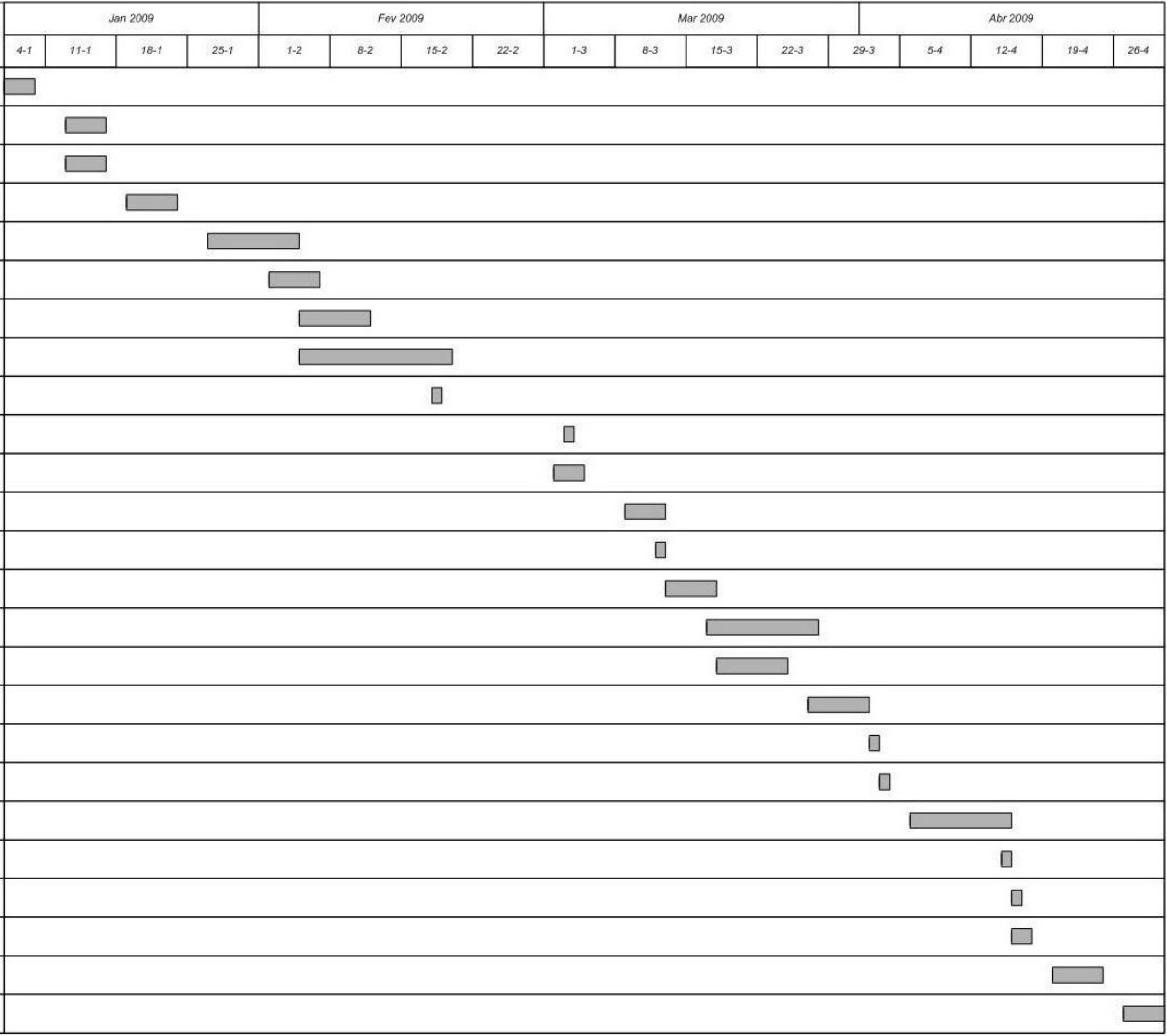
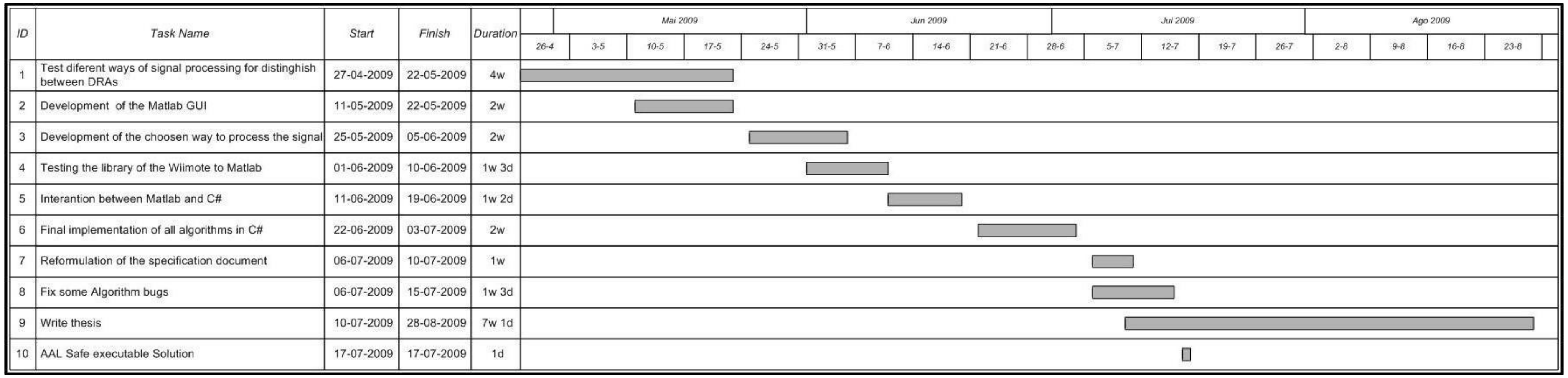


Table 4- Scheduling for second semester, part II.



ID	Task Name	Start	Finish	Duration
1	Test diferent ways of signal processing for distinghish between DRAs	27-04-2009	22-05-2009	4w
2	Development of the Matlab GUI	11-05-2009	22-05-2009	2w
3	Development of the choosen way to process the signal	25-05-2009	05-06-2009	2w
4	Testing the library of the Wiimote to Matlab	01-06-2009	10-06-2009	1w 3d
5	Interantion between Matlab and C#	11-06-2009	19-06-2009	1w 2d
6	Final implementation of all algorithms in C#	22-06-2009	03-07-2009	2w
7	Reformulation of the specification document	06-07-2009	10-07-2009	1w
8	Fix some Algorithm bugs	06-07-2009	15-07-2009	1w 3d
9	Write thesis	10-07-2009	28-08-2009	7w 1d
10	AAL Safe executable Solution	17-07-2009	17-07-2009	1d

3 RELATED WORKS

The aim of this chapter is to provide the reader with a perspective about the problems that Ambient Assisted Living Joint Program (AAL JP) together with its solutions intend to minimize. In a second part a description of state of art is done, showing some systems of fall detection. Finally a brief caparison between the *ALL* Safe and the other systems is done, showing the differences between them.

3.1 Problem Analysis

3.1.1 Ambient Assisted Living Joint Program

The ALL JP is a research and development funding program, composed of 20 actual members of the European Union, including Portugal, and 3 more associated states, Israel, Norway and Switzerland (1).

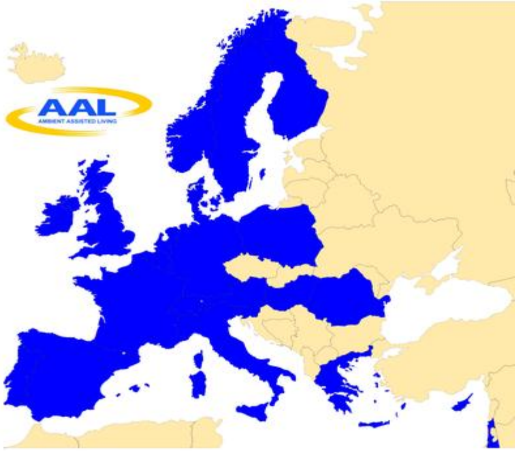


Figure 1- The 23 partner states of AAL JP (2).

This program was implemented in 2008, and it has a minimum duration of 5 years. Since the beginning, the program had two important events entitled “Call for Proposal” one in 2008 and the other in the present year. These events have different topics and goals, and their outcome is a call publication document, with all information about the program progress (1).

The topics are related with Information and Communication Technologies (ICT) solutions, and its applications in elderly people environment.

3.1.1.1 Why did AAL JP appear?

The ALL program appears due to the demographic changes that are occurring currently on the European society and the opportunity that these changes represent to ICT's. These demographic changes are related, mainly with the increase of elderly population.

Recent studies show that population, mainly in industrial countries, around the world is ageing. This occurrence results from two trends, the decrease of fertility rate and the increase of life expectancy. In the European Union the share of elderly in the population will double by 2050, this means that the Elderly Dependency Ratio (relation of the number of people aged 65 over the population aged 15-64) will double too (3) (4).

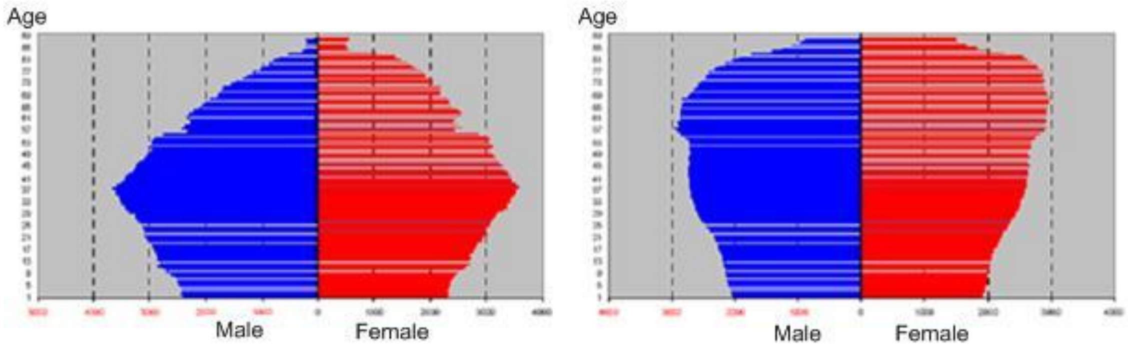


Figure 2-Age Pyramids for EU25 in 2004 and 2050.

In the same way, in 2050 the medium age in Europe will rise 10 years, from 38 to 48 (4).The consequences of these demographic trends reflect not only in the social field but also in the economical and financial field, for the reason that the Employment rate will decline sharply and financial sustainability of the welfare state will be at risk (3).

In Portugal the scenario is not different. The projections are for an increase of elderly dependency ratio and a reduction of young people dependency ratio; this will result in the decrease of sustainability potential ratio (relation between working population and elderly population) (5). The prevision for 2060 is 172 persons of working age for 100 elderly persons, less than half that in 2007 (5). The Elderly

Dependency Ratio in 2050 will be 0,581, more than the double on the one in 2005 that was 0,252 (4).

"While ageing is becoming a mainstream phenomenon, industry and providers do not yet sufficiently capture the needs of the ageing society in mainstream products and services" (6).

Both the elderly and their relatives care about this situation since well being is desirable. So, they wouldn't mind spending their savings in healthcare products and services. The considerable changes in demography are a great opportunity to ICT, because the elderly are 20% of the world population being, thus, a very important and potential market segment (6).

3.1.1.2 AAL JP's Objectives

The most important objective of ALL JP is to improve the quality of life of elderly people and strengthen the ICT industries' in Europe (1). To a better understanding of the AAL JP it is important to know its aims:

- To increase the time people can live autonomously in their favorite environment, their own house or a nursing home;
- To prevent loneliness and isolation, maintaining the multifunctional network around the individual;
- To maintain health and functional ability of the elderly people;
- To improve the security;
- To provide a healthier lifestyle for individuals at risk;
- Ensuring sustainability of health and social services in terms of financial human resources;
- Legal and ethical compliance, respect of the privacy and dignity of the elderly ;
- To support families and care organizations (1).

These aims are only achieved through the solutions that ICT provides. In this field it is important to foster the necessity of new ICT products, services and solutions for ageing well, at home in care organizations and in the community, thus reducing health and social care. Moreover the increase of conditions for industrial exploration of new researches enables the creation of a critical mass of research, development and innovation. All these achievements still enable the creation of jobs and business opportunities for European industry (1).

3.1.1.3 Our solution in AAL Topic Groups

The needs and opportunities of AAL could be represented in a diagram. In the core is the main goal the “Well being Person”. To achieve this goal the person’s environment must provide good conditions for people’s life. The “Well being Person” is connected with two different environments, the “Person@Home”, where the person spends more time and physical well being is more relevant, and the surrounding environment where the social interactions like mobility and Working Life are the greater preoccupations.

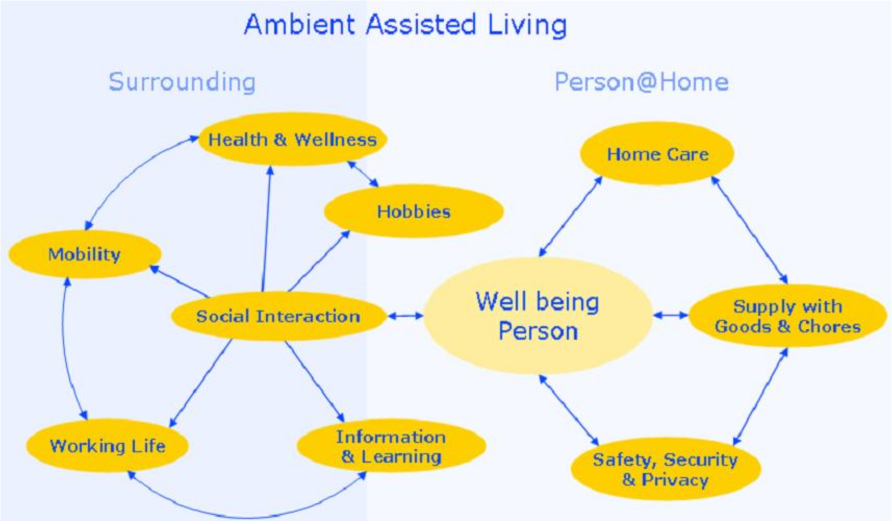


Figure 3- AAL Topic Groups

Our solution belongs to the “Person@Home” environment and intends to improve the safety and security of the elderly people, alerting when a fall happens and providing instant help.

3.2 State of Art

Through the following State of the Art, it is intended to summarize what solutions others have developed for minimize the above referred problematic situations.

All the systems presented in this section were created with the objective of remotely monitor patients’ falls and activities.

3.2.1 *iLife* Fall Detector Sensor

The *iLife* is a system that provides automatic fall detection with sentinel, and activity monitoring. The system is composed of microprocessors and multiple accelerometers. *iLife*'s proprietary fall detection and methodology is described in US Patents 6,703,939; 6,661,347; 6,501,386; and 6,307,481.

This device is totally compatible with the complete line of AlertOne technologies, which allows professional home care 24h a day.

The *iLife* is capable to distinguish between falls and normal movements that involve shocks like fast walking and descending stairs. It is also programmed to identify long periods of rest and call for help. This alarm can be activated when the user has forgotten to put it on or intentionally removes it.

The device has an inconspicuous and attractive design, which incorporates the sensors and the wireless radio transmitter (7).

Figure 4- *iLife* device design.

It is composed by a button to allow call for help automatically, a LED to confirm the visual operation of the system and a belt clip.

The sensor is a two-axial accelerometer, and it is able to identify the inertial acceleration and the static acceleration (gravity). The sample rate may vary, depending on the positions and the level of activity of the user. The most relevant technical details of the *iLife* system are presented in the following table (7).

Table 5-Technical details of the *iLife* System.

Technical details	
Associated Products	ResponseAlert IdentiFind ResidentCare
Dimensions	22.5x27.5X5.7mm
Weight	≈ 57g
Battery Life	3 to 6 months based on end-user level of activity

3.2.2 Falls Management Solution

The Falls Managements Solution is a system developed by Tunstall group. The system is constituted by a range of non intrusive sensors, which provide constant monitorization of the risk of fall, differently or indirectly. All these sensors are connected with Lifeline home whereby the instant help can be called. The telecare sensors are:

- **Fall Detector**- the Fall Detector was developed to be used in the belt or in a pouch around the waist. It uses two different processes to identify the falls, the impact and the tilt angle of the user. When an impact happens and if subsequently the user is not in vertical position, the system emits a buzzing noise to alert him that an alarm call is raised. If the user is ok the alarm can be cancelled by pressing the button (8).

Figure 5- Fall Detector design.

- **Bed/ Chair Occupancy Sensor-** this device was developed thinking of night time, when the fall detector isn't worn. This sensor is placed under the mattress, and can detect when someone leaves the bed, turns on the bedside light to minimize the risk of fall. The sensor can be programmed to raise an alarm if the user hasn't returned to bed in the stipulated time, avoiding a long period without help, if the fall has occurred. Similarly this device can be used on chairs and wheelchairs (8).

Figure 6- Bed/ Chair Occupancy Sensor

- **Passive Infrared Detector-** This kind of sensor is specialized in detecting movements is a restrict area. This sensor is integrated in the Falls Management Solution with the goal of raising an alarm if it doesn't sense movement in a room for a long period of time (8).

Figure 7- Passive Infrared Detector.

- **Pull Cord-** The Pull Cord finds its greatest utility in bedrooms, it can be installed in about any place in a room, so it can trigger an alarm call when a person falls and is not wearing the fall detector.

Figure 8- Pull Cord.

The Falls Management Solution may incorporate all these sensors, but the most relevant is the fall detector. In the following table its technical details are shown (9).

Table 6- Technical details of the Fall Detector.

Technical details	
Associated Products	• Tunstall Lifeline 400 • Tunstall Lifeline Connect +
Dimensions	75x53X28mm
Weight	75g
Battery Life	6 months
Range	250m

3.2.3 myHalo

MyHalo is a monitoring system developed by a healthcare company entitled Halo Monitoring. The MyHalo system was developed thinking of the elderly and their caregivers. MyHalo is capable of monitoring vital signs, activity, asleep and wake patterns, being the most interesting feature of the system its detection falls ability and capability to call for help instantly (10).

The system is composed of one chest strap, which incorporates the sensors and one transmitter. A home gateway, that communicates with the transmitter wirelessly. The home gateway is connected to the health server that controls the flow of information, and it can send text messages and emails directly to the caregiver or to a call center. The health server is also responsible to manage the web page where all the information about the user is available and, when possible, see his/her monitorization on line in real time. The following picture shows the entire architecture of the system (10).

Figure 9- Architecture of the myHalo system.

The chest strap contains the sensors capable of measuring user orientation and motion. The data is constantly processed in real-time, if a fall occurs the transmitter sends a message to the gateway indicating a fall. The system differs between periods of activity and rest. So as to make the system register the rest periods, the user has to be lying. The chest strap has an agronomic design and was developed to be used 24 hours a day. It has a battery life of one year (10).

Figure 10- Chest strap.

The grey button is designed to provide automatic emergency response, when the user feels the need to call for help. When the button is pushed an alarm is sent and the caregivers are informed, the same happens when a fall occurs (10).

3.3 Review of our System

In this section a comparison will be made, between the systems mentioned previously and the *AAL* Safe, in terms of their technical details/design and capabilities.

3.3.1 Technical details/ Design

In the next table it is possible to see the main aspects related to the design and technical details of the systems sensors.

Table 7- Comparison of technical details of all systems.

System	Dimensions	Weight	Battery life	Range
	146x24x29mm	129g	60 hours if used alone and about 25 if using accelerometer	10m
	22.5x27.5X 5.7mm	≈57g	3 to 6 months based on end-user level of activity	___
	75x53X28mm	75g	6 months	250m
	___	___	1 year	___

In terms of technical details the Wiimote shows some limitations, its design is not the most appropriate for a fall detector, and the battery life and range is reduced compared to the other systems. However this doesn't represent a big limitation since this project is a concept proof that use the Wiimote and its accelerometer develop a fall detector and DRAs identification. In a final version of the AAL Safe system the Wiimote will be replaced by a more appropriate feature, developed thinking exclusively of a fall detector.

3.3.2 Capabilities

All systems monitor the patients in real time and have the ability to send alarms and call for help instantly. The *iLife*, the Fall Detection Solution and myHalo systems have a phone line associated to send the alarms, in the AAL Safe system the help is requested locally since the system was developed to think of a care institution. The most relevant capabilities of all systems are compared in the following table.

Table 8- Comparison of capabilities of all systems.

Systems / Capabilities	*AAL* Safe	*iLife*	Fall Detection Solution	myHalo
Detect Fall	✓	✓	✓	✓
Button to avoid fake fall	✓	✗	✓	✗
Panic Button	✗	✓	✗	✓
Differentiation between rest and activity	✓	✓	✗	✓
Tilt angle	✓	✗	✓	✓
Energy Expenditure	✓	✗	✗	✗
Differentiation between lying, sit and stand	✓	✗	✗	✗

All the systems have the capability to detect falls and have a button, in the case of *AAL* Safe and Fall Detection Solution this buttons is used to avoid fake falls, and falls where the patient has fallen, but doesn't get hurt. In the case of the *iLife* and myHalo systems the button is used to call for help instantly. This is the unique ability that the *AAL* Safe system doesn't support, which is easily incorporated due to the wide availability of programmable buttons that the Wiimote has.

Regarding the differences between periods of rest and activity the ALL Safe uses the Signal Magnitude Area (SMA), while the myHalo assumes that the lying position calculated from the tilt angel, represents the periods of rest. In the iLife system the way to differentiate between activity and rest is not known. This capability is used to detect longs periods of rest and send an alarm.

The tilt angle calculation is a capability supported by the AAL Safe, myHalo and Fall Detection Solution, in the Fall Detection Solution this capability is used to check the position of the patient after the possible fall. In the AAL Safe the tilt angle is used to distinguish between sit, stand and lying positions which is an exclusive capability of the AAL Safe, as well as the calculation of the energy expenditure.

4 REQUIREMENT ANALYSES

In this chapter an overview of the Wii console and its peripherals is given. This provides some details about Wiimote hardware and its capabilities, mainly the features of interest for our project. An overview of the available software Application Programming Interfaces (APIs) is presented, as well an analysis of the library and programming language selected for this project.

4.1 Wii Console and Peripherals

The Wii is a most revolutionary video game console, and was created by Nintendo. The Wii console not only created a revolution on the video game market, but also in the way to play. Its wireless nature and the realistic game scenes, gave it all this success that incited academic students and researchers to explore its capabilities.

The Wii console has the ability to support several Wiimotes connected by Bluetooth, allowing up to four players connected at same time. Connected to Wiimote is a sensor bar that contains 10 IR LEDs. The LEDs are positioned with the intention of increasing the sensing range of Wiimote’s IR camera (7). The Wiimote is the major peripheral of the Wii console, and the others communicate through it to the console, except the Balance Board. The other peripherals are the Nunchuk, the Classic Controller, the Balance Board, Wii MotionPlus and more recently appeared the Vitality Sensor (7) (8).

- **Nunchuk -** this peripheral connects to the Wiimote through a cable or via Bluetooth in more recent versions. The Nunchuk contains two buttons; a TA accelerometer, with 3 degrees of freedom (DOF) and an analogue stick (7). The function of the Nunchuk is to provide to player with more realistic game scenes through the interaction of the two acceleration data, from itself and Wiimote.

- **Classic Controller -** connects to Wiimote via a cable; contains two analogue sticks and a number of input buttons (7) .

- **Balance Board -** this peripheral is the only that does not depend on Wiimote, it communicates directly to the host controller via Bluetooth. The Board Balance contains four pressure sensors, which are used to calculate the user's center of balance and body mass index. This peripheral was developed specially to the Wii Fit game (7) .

- **Wii MotionPlus** – this is the last peripheral that was incorporated in the Wii game. The Wii MotionPlus affixes directly to the Wiimote's bottom side, increasing its size. This peripheral incorporates a gyroscope, which associates with on accelerometer and a sensor bar that enables a better performance of the Wiimote's motions capabilities. The gyroscope is able to determine rotational motion and it is also called of angular rate sensor. The Wiimote accelerometer and Wii MotionsPlus gyroscope associated to each other, allow the capture of more complex and accurate movements with 6-DOF (13).

- **Vitality Sensor** – the most recent peripheral of the Wiimote is the Vitality Sensor, like the Nunchuk it connects with the Wiimote by cable. It is a finger sensor to read the pulse, the idea is to use this oximeter to check the stress and use this parameter in games where to detect a lie is the goal, for example (8). The Vitality Sensor may represent a tool, which can be interesting to the Biomedical Engineering.

Figure 11- Wii console's peripherals; Nunchuk; Classic Controller Pro; Balance Board; Wii MotionPlus and Vitality Sensor respectively.

4.2 Wiimote Hardware

The Wiimote, as already mentioned, is a primary peripheral of Wii console, for this and for the reason that is the only tool necessary for the development of this project, it deserves a special attention. The Wiimote is recognized as Human Interface Device (HID) input device, with 6 – DOF, this means that through the Wiimote, it is possible to manipulate objects in 3D. Three of these degrees are obtained by the translation of X Y and Z axes, and the others are obtained by the X Y and Z rotation (9). In the Wiimote these 6-DOF are obtained by the interaction of the accelerometer and the IR camera. (10)

Figure 12 - Wiimote viewed from different perspectives.

The Wiimote is composed of a single-chip Bluetooth, an IR camera, an accelerometer, four LEDs, twelve buttons and also an embedded speaker and a rumble feature. For this project the accelerometer and the Bluetooth are the most relevant hardware components of Wiimote, and so are some buttons and the rumble feature.

- **Chip Bluetooth** – the chip BT is a Broadcom 2042 driver device with fully integrate HID profile and full BT stack. It has on-board a 8051 processor and ROM/RAM memory. The chip belongs to class 2 BT devices. The distance range is approximately 10m; it is the standard to class 2 BT devices. This chip is appropriate to mouse and keyboard applications, but also to remote control HID, which is the case of Wiimote (11).

- **IR camera** – the IR camera is located at the front of Wiimote, which is shown on the right side of Fig-12. The camera has a resolution of 1024x768 for the tracked points at 100 Hz and contains an IR filter attached that senses light within near-infrared spectrum, with wavelengths greater than 800nm.

Without the filter the camera can track any bright object, like incandescent lamps, candles and daylight, decreasing the precision of the camera (7).

- **LEDs –** the Wiimote has four LEDs located at the bottom edge. When the LED's are blinking it means that the remote is in discoverable mode, ready to be connected. During the Wiimote's applications the LEDs are used to show the battery level (7).

- **Buttons-** the buttons are spread across the Wiimote, four of them are arranged directionally, they have a cross shape. The buttons 1 and 2, when pressed simultaneously allow the remote to get in discoverable mode and the LEDs blink for 20s. During this period it is possible to connect the remote to a Bluetooth HID driver on the host. If the connection is not established during the period, the Wiimote will turn off (7).

- **Rumble feature –** the rumble feature is composed of a motor attached to an off-centre weight. The motor attached on the remote is a *SEM8728DA.*This functionality is used to provide feedback during the game play and is activated via Wiimote API on a PC (12).

- **Speaker-** the speaker is also attached on the Wiimote, it is a small low-quality piezo-electric and its functionality is to provide sound effects during the game play. The library used in this project doesn't support this functionality (12).

- **Memory -** The Wiimote contains a 16 KB EEPROM chip from which a section of 6 kilobytes can be freely read from and written to by the host controller. This allows an easy storage and transportation of custom application data. Certain hardware and software calibration settings, as well as interface options, can be stored in the Wiimote by this way and this data can be accessed anywhere by an appropriate host (7).

- **Accelerometer** - The accelerometers belong to a group of Inertial Mounting Units (IMUs) sensors. This kind of sensors aim to measure the orientation and position of the unit through space and time.

An accelerometer measures the acceleration along an axe, in the case of TA accelerometer, it measures the acceleration along the three axes, X, Y and Z. The linear accelerometers are a kind of accelerometers that usually consist of springs of mass systems that are exerted by gravitational forces, which result from the earth gravitational field (g=9.81 m/s^2), and bodily motions, stocks or vibrations (9).

The motion of the Wiimote is sensed by a TA linear accelerometer, an ADXL330, manufactured by Analog Devices, which has a measurement range of ± 3 g with 10% sensitivity. The sensor is built on top of silicon wafer, which is suspended by polysilicon springs. This silicone structure provides resistance against acceleration forces. The displacement of the structure is measured by a differential capacitor and converted in voltage. This voltage is proportional to the acceleration of the structure and then the acceleration is digitizing (7) (13).

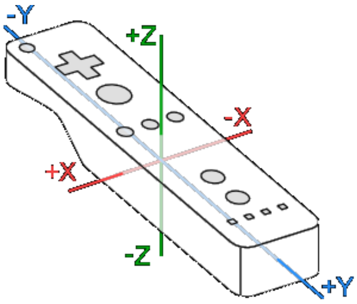


Figure 13 - Wiimote with the directions of accelerometer.

When dropped, the Wiimote gives an acceleration of approximately 0. The defect of manufacture and calibration really cause some intrinsic zero offsets, but this situation is prevented because the chip is self-calibrated during manufacture to produce offset values for any measurement defects present. When the Wiimote's flash RAM stars, these calibration values are stores. (7)

The most important specifications of ADXL330 linear accelerometer can be seen in the following table.

Table 9- ADXL330 accelerometer´s specifications.

Characteristics	ADXL330
Range [g]	±3
Sensitivity[%]	10
Full scale [g]	10
Linearity [% full scale]	0.3
Noise density [μg/$\sqrt{\text{Hz rms}}$]	
X_{out}, Y_{out}	280
Z_{out}	350
Dimensions[mm^3]	4x4x1.45
Alignment[°]	±0.1
Sample Frequency[Hz]	100

4.3 Bluetooth Communication

Bluetooth is wireless communication which is characterized by its low cost and short range radio technology, depending on the class that is assigned. Its main potentialities are:

- Low size
- Operation in unlicensed band, 2.4 Hz
- High security of the encryption and authentication
- Defense against interference
- Point-to-point and point-to-multi point communication
- Low power consumption

Bluetooth technology is divided in to three classes according to the power and range as shown in the next table:

Table 10 – Classes of Bluetooth

Class	Power	Range
1	100 mW	100m
2	2,5 mW	10m
3	1 mW	1m

The range may vary according to the environment, physical barriers may limit the range of the communication (14).

The fundamental form of Bluetooth communication is created on the basis of masters/slaves, creating physical links between them. A master can control up to seven slaves in his area. All these devices in communication form a small network called Piconet and the physical links between them are characterized by a Radio Frequency combined with temporal parameters. The master is the first device connected and is the one that sets the clock, frequency jumping sequence and the access code for the link with the slave. The data transmission is based on principle of Time Division Duplex. At first the master sends a package to a specific slave, then, in the following time interval the slave response is sent to the master (15) (16).

The Wiimote is a class II Bluetooth device, in our system it is the slave and the PC represents the master. In other words, the Wiimote is an HID that transmits and receives data from a host, in this case a PC. As all Bluetooth devices the Wiimote has a descriptor block, which can be accessed through the Service Discovery Protocol (SDP), at the beginning of the communication. The SDP allows the access of device services, characteristics and, subsequently, the connection between two or more devices may be established. When queried by the SDP, the Wiimote reports the information shown in the Table 13. This information is unique to the Wiimote and distinguishes it among all other registered Bluetooth devices (7).

Table 11 - Internal characteristics of the Wiimote.

Attribute	Data
Name	Nintendo RLV-CNT-01
Vendor ID	0x057e
Product ID	0x0306

The descriptor block includes also an enumeration of reports that is transmitted to the host, which report IDs and payload sizes for the bi-lateral communication link between the host and the Wiimote. The table with the detailed HID descriptor blocks can be consulted in Annex A.

4.4 Wiimote API

The programming language selected to realize this project was the C#, considering this, the choice of the API is more limited, and, in the moment of the choice the only available option was the WiimoteLib. This library was developed by Brian Peek, and is available in two versions, one for C# and the other for VisualBasic.NET (17).

With the WiimoteLib comes a test program in which it is possible to observe all Wiimote´s functions. The library not only supports the interfacing with the Wiimote but also with all this peripherals and the last version already includes the Wii MotionPlus. The next peripheral to be included in this library will be the Vitality Sensor.

The way of work with the WiimoteLib, and their main commands, can be consulted in Annex B.

5 SYSTEM ARCHITECTURE

The goal of this chapter is to provide a detailed overview of the system architecture. The chapter is divided in to two sections. The first one describes the Physical Architecture, which is related to the system features and the environment where they are placed, such as human interaction with them. The other section describes the Logical Architecture i.e. the process and the data flows.

5.1 Physical Architecture

The AAL Safe is a system that has a double interaction, by the patient and by the user. As it can be seen in the figure below, the system was developed thinking of a care institution. Each patient (1) has attached to himself a transmission module, the Wiimote, which monitors his/her, daily activities and detects possible fall occurrences. This information is sent to a PC, the acquisition module, where a healthcare responsible (2) checks the monitorization. The transmission module is portable and accompanies the patient for wherever he/she goes, inside the care institution environment.

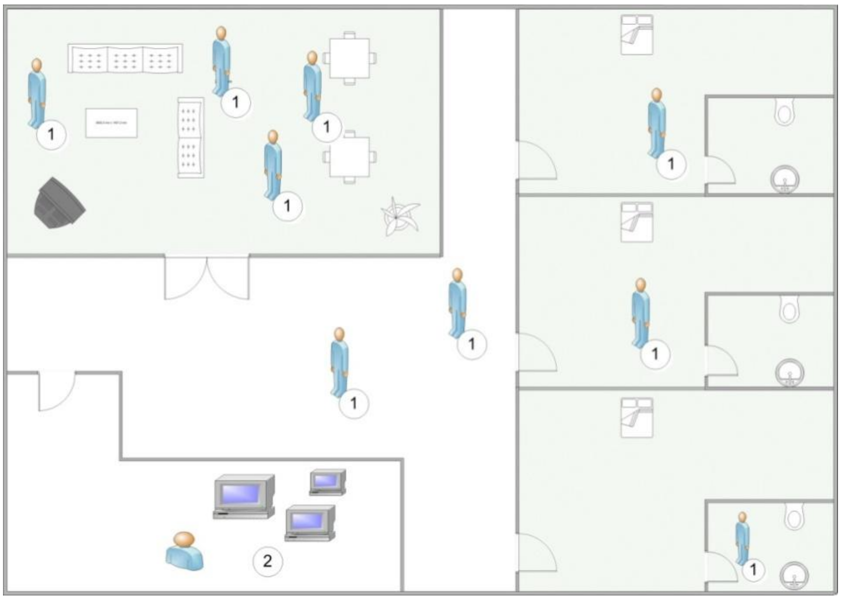


Figure 14- Physical Scheme of *AAL* Safe

The Physical Architecture is composed by two main parts, the Transmission Module, and the Processing Host that is also responsible to acquire the data. The Transmission Module is embedded in the Wiimote which is attached to the patient. The data transmission is made via Bluetooth, from the Wiimote to a PC in real time. The PC has an Application that contains three main modules: Database, Reports and Processing data.

In a care institution environment and taking into consideration the Bluetooth protocol; one PC with several dongles are enough to monitor the patients, being possible to connect seven Wiimotes for each dongle Bluetooth. Other parameter to take into account is the Bluetooth class of the host, which must be equal or higher than the Wiimote, not to limit the range of interaction between the two devices.

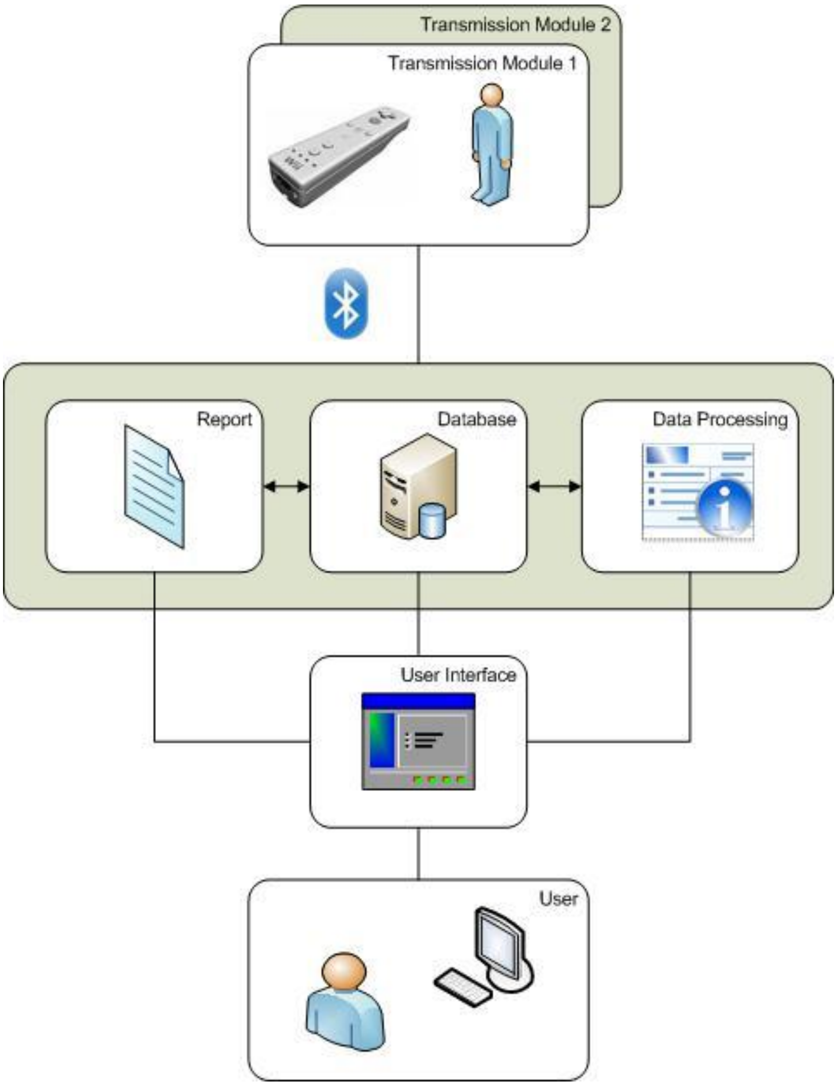


Figure 15- Physical Architecture of the *AAL* Safe.

5.2 Logical Architecture

The *AAl* Safe solution presents a logical architecture with multi tiers. The first tier is entitled the Presentation Tier and contains the Windows Application, being responsible to coordinate the application flow; this tier allows all the interaction between the user and the system. The Logical Tier is the second tier, and it incorporates the report module, the processing data module and the transmission module. The third and last tier contains the Database, which is responsible to store all the information about the patients and their monitorizations.

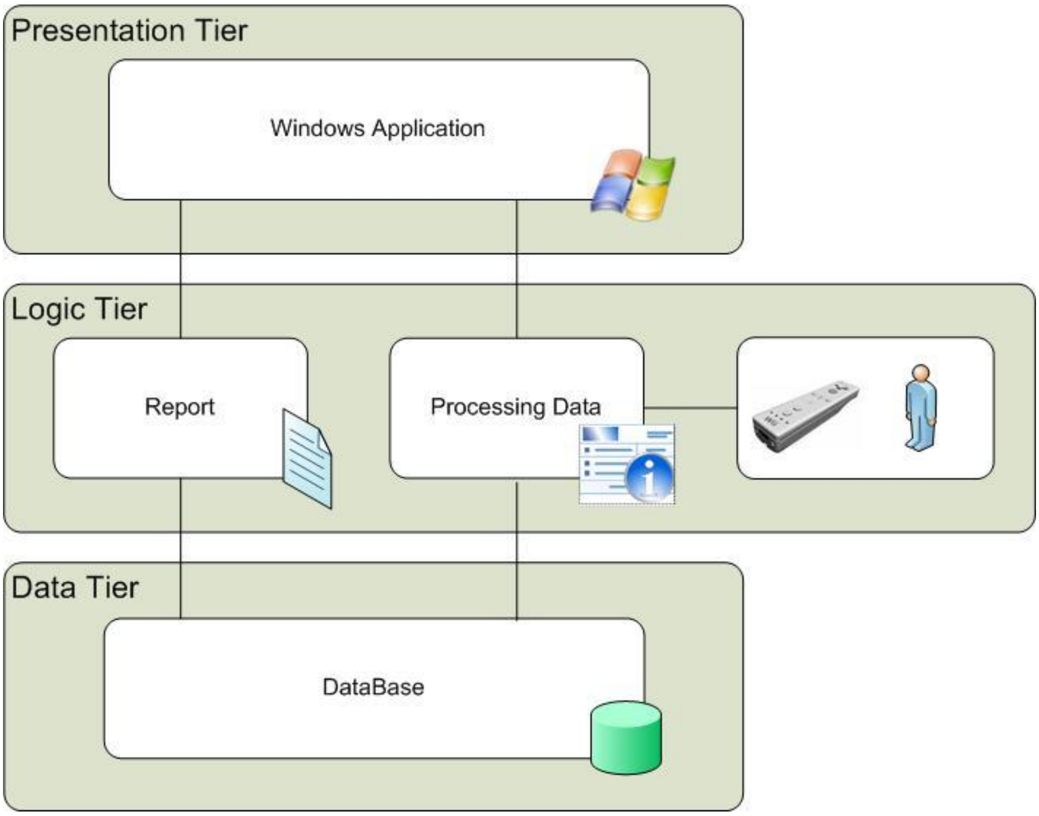


Figure 16-General Logical Architecture

The most important advantage of this architecture is the possibility the Processing Data module has being able to work autonomously, which means that while one or more patients are being monitored, it is possible to access the other modules and handle the other options and functions of the Windows Application. This advantage is better seen in the following diagram, which gives a better understanding about the process flow between the modules of the architecture.

The Windows application is the responsible to coordinate all the processes. Using the Windows Application it is possible to access to patient details and past session details, inserted in the database. Furthermore, through the Windows Application it's possible to request a new session. A session is responsible to monitor the patient in real time. To begin a new session the transmission module must be connected to the host, this is an independent process that must be done before the request for the new data processing. When the new session begins the Processing data module is able to detect falls and distinguish the DRA's. At the end of the session it's possible to generate a report with all the information about the patient and the session. The report can be requested from the database, which contains the sessions details associated with each patient.

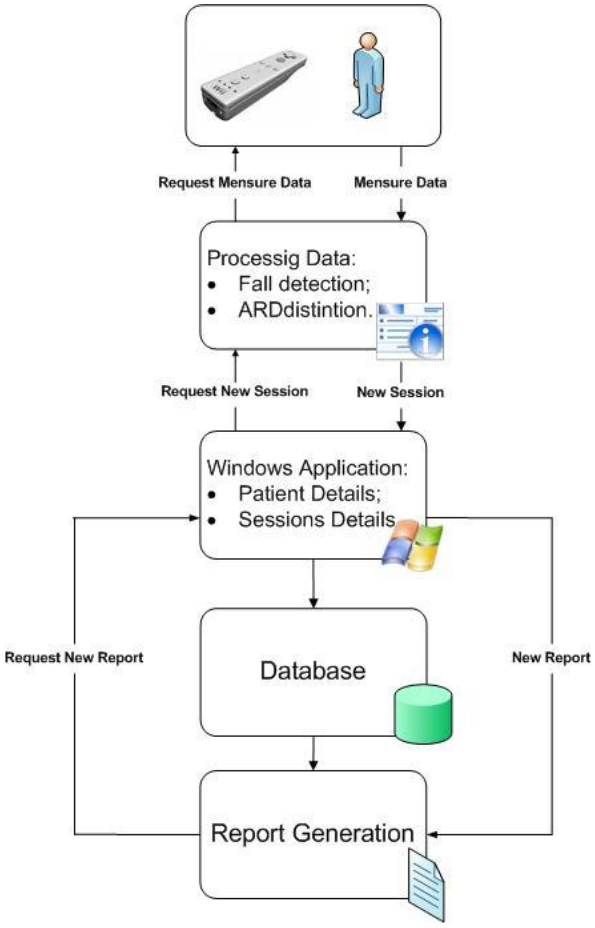


Figure 17-Logical Architecture

6 SIGNAL PROCESSING

In this chapter will be described the methods tested and used to process the acceleration signal, to identify desired parameters, like DRA's and falls.

6.1 Signal Processing Tested Methods

The signal obtained directly from the Wiimote, without any treatment is not enough to identify and detect DRAs and falls accurately. It begins testing basic ways of signal processing, like power of signal, energy of signal and entropy. Comparing the original signal and the signal obtained by these methods it is concluded that they don't represent a good way to identify the desired parameters.

The Fast Fourier Transform (FFT) was also tested and represents a tool with better results in classification of dynamic activities (18). However this method was rejected for the reason that its ability is to identify dynamic activities like walking, cycling, running etc. As the objective of this system is to identify the posture of the patient inside the home environment, the FFT method was rejected.

6.2 Signal Processing Used Methods

The method chosen to identify the postures is based in filtering. Two filters were applied to obtain the desired signal. Firstly a median filter was applied and then a Low Pass Filter (LPF) was applied. These two filters are digital, this means that they operate in discrete-time signals as it is the case of the acceleration signal. The output of these two filters is used to implement the algorithms decisions (19).

6.2.1 Median Filter

The median filters are quite common in image processing but also useful in one-dimensional case like the accelerometer signal. This kind of filter belongs to the non-linear filter class and aims to smooth the signals and to suppress the noise, they are also filters that act in time domain. The main advantages of the non-linear filters are their capability in remove the effect of impulsive noise and preserve the edge (20).

The median is given by the equation (1):

$$med(x_i) = \begin{cases} x_{(v+1)} & n = 2v + 1\ ,\ n\ odd \\ \frac{1}{2}\ (x_v + x_{v+1}) & n = 2v\ \ ,n\ even \end{cases} \qquad (1)$$

Where *n* represents the number of observations x_i ,i=1,…,*n*, and *v* represents the one position inside the interval of observations.

In the one-dimensional case the filtering process consists of sliding a window of numbers along the signal. If the window has an odd number of elements, the centre sample are replaced by the median of the samples in the window. If the window has an even number of elements, the tow samples on the middle are replaced by the average between them (20) (21).

The median filter used in this project is an order 3 filter, this means that the windows size is equal to 3 and *n*=3. The input-output relation of this filter is given by the equation (2), where the y_i represents the output signal, and the x_i represents the input signal (19) (20).

$$y_i = med(x_{i-v}, \dots x_i, \dots x_{i+v}), \qquad i \in \boldsymbol{Z} \qquad (2)$$

This filter is applied to remove any abnormal noise spikes produced by the accelerometer.

In the following picture it is possible to see the signal from the Z axis before and after the filter application.

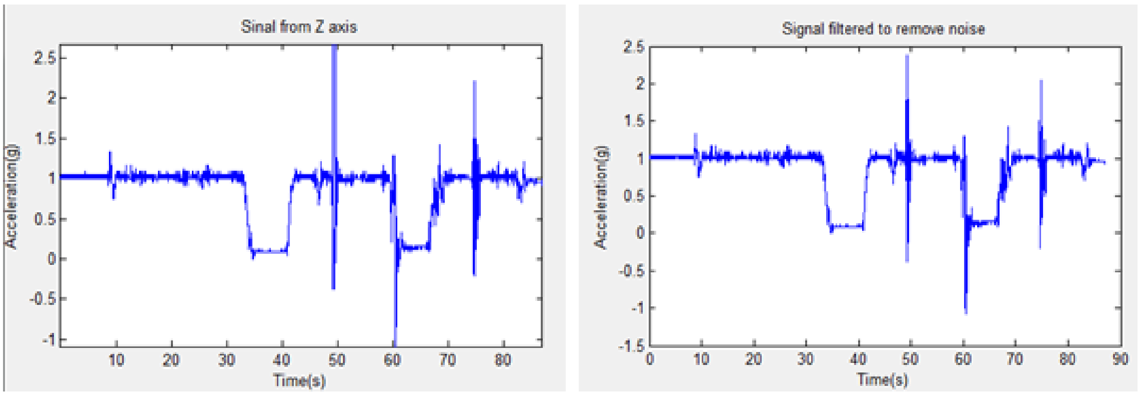


Figure 18- Signal before and after of the median filter application.

6.2.2 Low Pass Filter

The LPF is the response type of the digital filters that acts in frequency domain. Beyond the response type this digital filter is also characterize by impulse response duration.

In terms of response type, the LPFs are characterized by attenuating signal frequency above the cutoff frequency. The cutoff frequency is the frequency above which the power in output of the signal is reduced to half the frequency of band pass. To this reduction correspond an attenuation of the -3dB and a reduction of the 70,7% in amplitude.

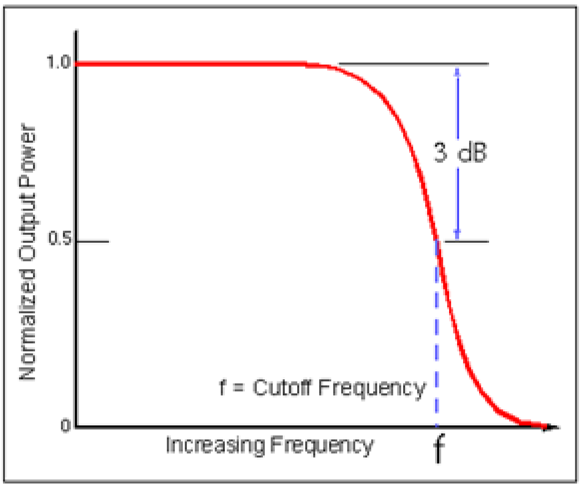


Figure 19- Typical frequency response of a LPF. **(25)**

To project the filter is used the Z transform, which is used specially to digital signal processing.

Regards to the impulse response duration the LPF is an Infinite Impulse Response (IIR) filter. These filters are similar to the analogical filters, they have the advantage of low computational complexity, and their major disadvantage is their nonlinear phase response, that is the relation between the input and output signal (22). The IIR filters can be divided in several types and the elliptic filters are one of them. They are characterized by havening ripple in pass band and stop band, and the sharpest cutoff frequency.

The filter used in this project is a LPF elliptical IIR filter; its parameters are shown in the following table (19).

Table 12- LPF elliptical IIR filter parameters'.

Parameters	
Order	3
Sample frequency	100Hz
Cutoff frequency	0.25Hz
Pass band ripple	0.01dB
Stop band ripple	100dB

This filter was projected in Matlab and its frequency and phase response are shown in the Figure 20.

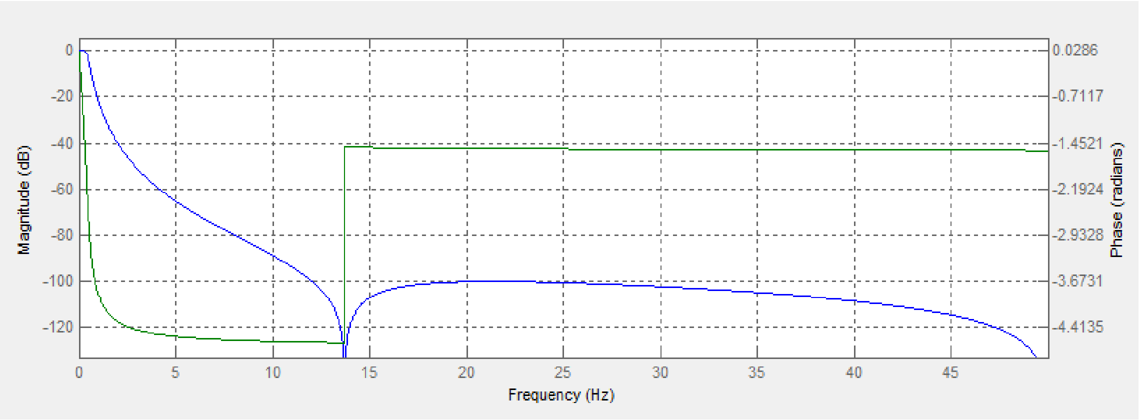


Figure 20- Filter's phase and frequency response chart.

It is possible to see that the nonlinear phase response is stronger in the beginning of the pass band, this may represent a disadvantage in signal accuracy, however the sharpest cutoff frequency typical of the elliptic filters, represents a good choice due to the complexity in separating the different components of the acceleration as well as the low computational complexity of this filter once the signal is processed in real time.

The LPF is applied at the output of the median filter, which is already without noise.

The purpose of applying this filter is separate~~s~~ the Direct Current (DC) and Alternate Current (AC) components. In terms of acceleration the DC component is related to acceleration due to the gravity (GA) and the AC component is related to the acceleration due to the body movements (BA). The signal output of the filter application is the GA component, which can vary between -1g and 1g (19) (23).

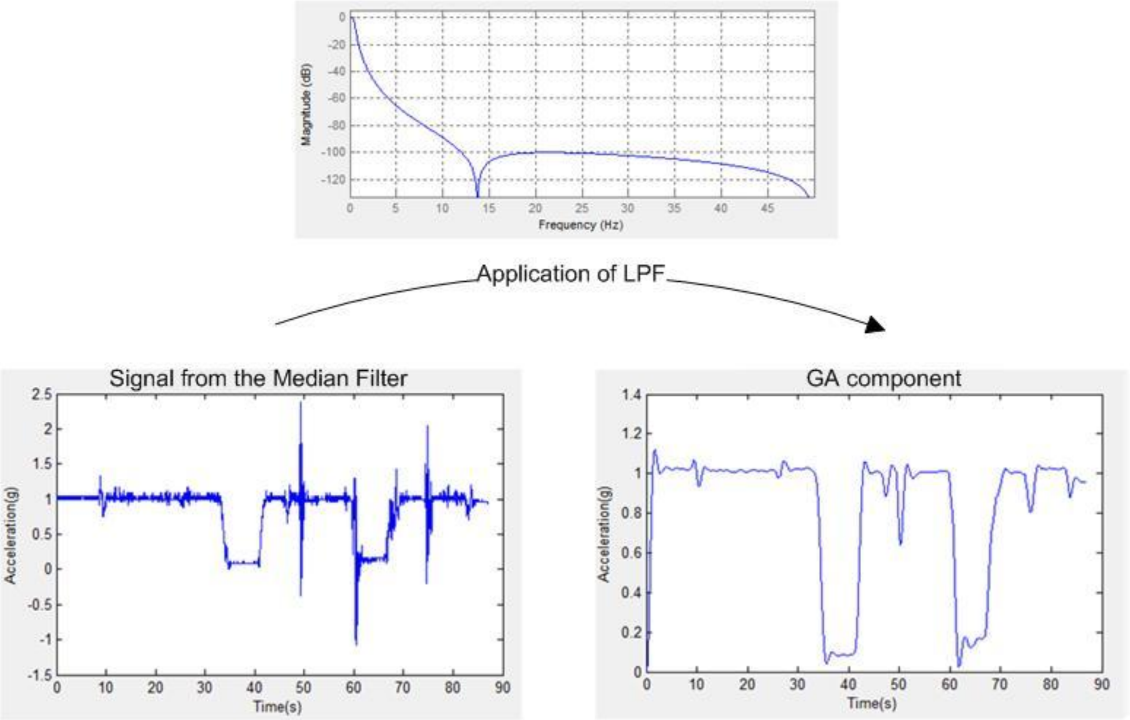


Figure 21-Signal before and after the LPF's application.

The BA component is obtained through the difference between the original signal, from the output of the median filter, and the GA component. These two components of acceleration have different applications in the algorithms of fall detection and DRAs identification, and they will be described in the next chapter.

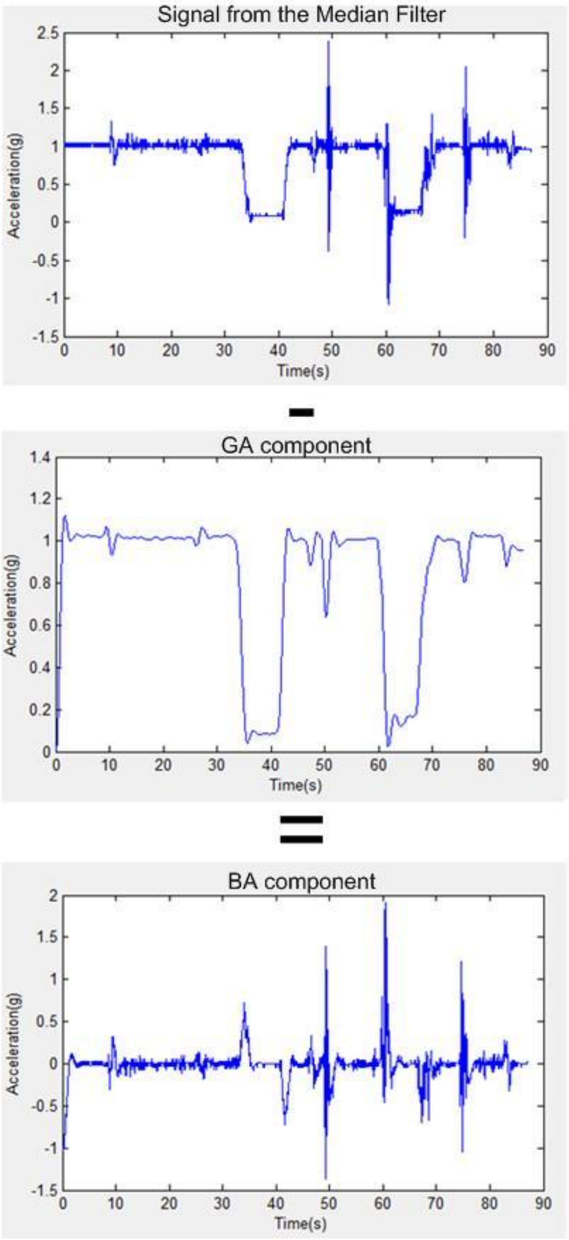


Figure 22- Obtaining of the BA component.

The LPF used presents two disadvantages, it takes approximately 1.5s to stabilize, and the values on the right side of the green line shown in Figure 23 have to be ignored due to its irrelevance. The other limitation is related to the fact that the GA component overcomes the expected limits, this is evident in the Figure 23 above the red line. These two disadvantages can be circumvented by programming without interfering in algorithm's decisions performance. This process will be described in chapter 8.

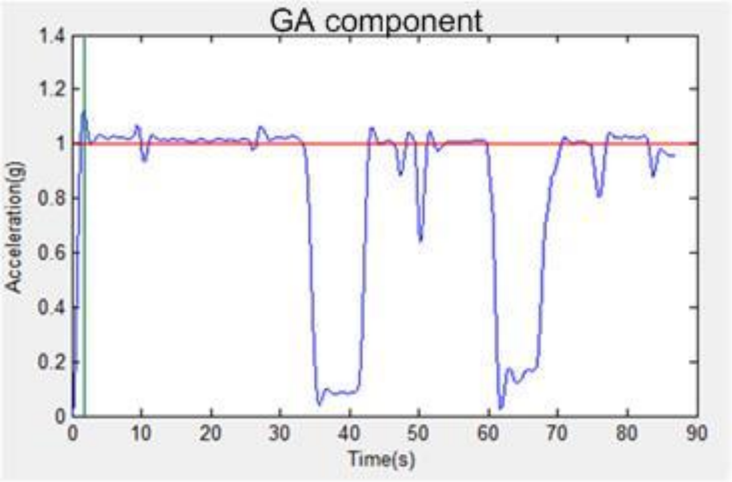


Figure 23- LPF’s limitations highlight.

7 ALGORITHMS

In this Chapter will be described the fall detection algorithm and DRAs' distinction, as well as all the decisions and parameters used in the algorithms development. Furthermore will be described the calculation method for energy expenditure based in acceleration signals. All algorithms developed are only based in acceleration signals.

7.1 DRA Distinction Algorithm

Even if it has begun by detection fall algorithm development, at the moment it makes more sense to start with the explanation of the DRA algorithm since the detention fall algorithm is incorporated in the DRA distinction algorithm.

7.1.1 Algorithm Type

This type of algorithm is based on an hierarchical binary tree; it means that each node of the binary tree has at most two descendants. This kind of algorithm structure is characterized by the general classifications on the top levels of the tree. These classifications have a higher level of certainty, and moreover, in lower levels of the tree the classifications are more detailed and have a lower level of certainly, for the reason that the methods used to identify them are more complex and less accurate. The categories of the same level are always independent and they cover all possible case of the parent category. The Figure 24 represents this kind of structures with two hierarchical levels, which support the fallback case. The fallback case represents the category that is accepted when all the other possible classifications are rejected. In the Figure 24 the fallback case is presented in the "other movement" and by the "sub-movement 2". In other words the fallback case represents the last option of each hierarchical level (24).

The advantages of using binary decision trees are: capacity of flexibility allowing the easy add/ remove categories, without affecting the other parts of structure the; speed processing increase and ensure that no valid logic paths are omitted (24).

The movements can be ordered from the most to the least probable, to minimize the processing, or can be ordered considering the critical movements first.

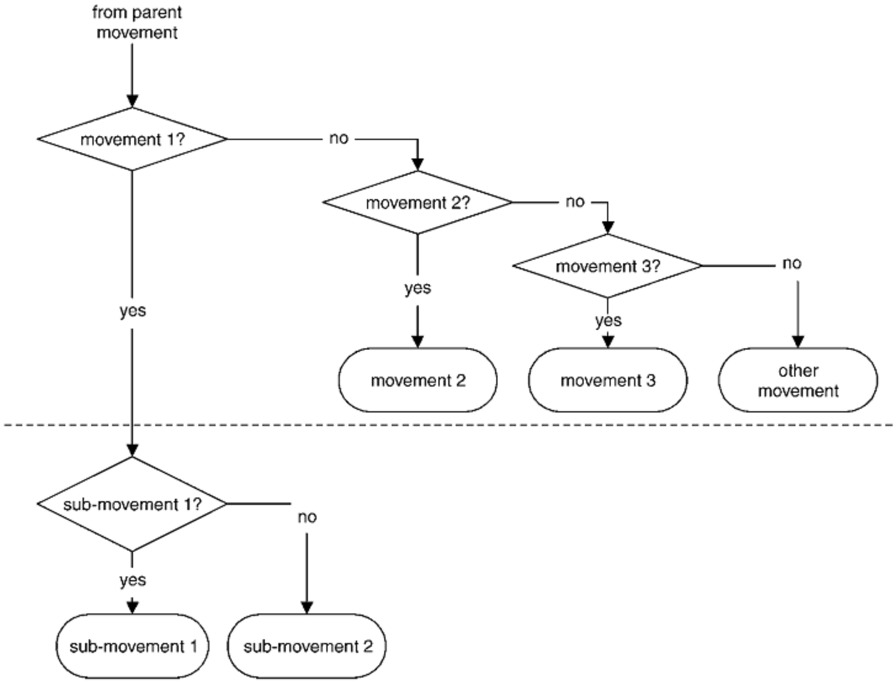


Figure 24-Example of a structure binary tree, the dotted line show different levels of the hierarchy.

7.1.2 Movements and Postures that Algorithm Intends to Identify

The next step to develop the algorithm is to decide which movements and postural orientations are intended to identify. The decisions are divided into four levels, the first level includes the acceleration signal and all processing and calculations required to distinguish between the different postures and movements. The first decision to make is distinguishing between periods of activity and rest; these two options are inserted inside level two. The activity is associated to movements, it could be a possible fall, upright active, transitions, lying active and the most unusual inverted active.

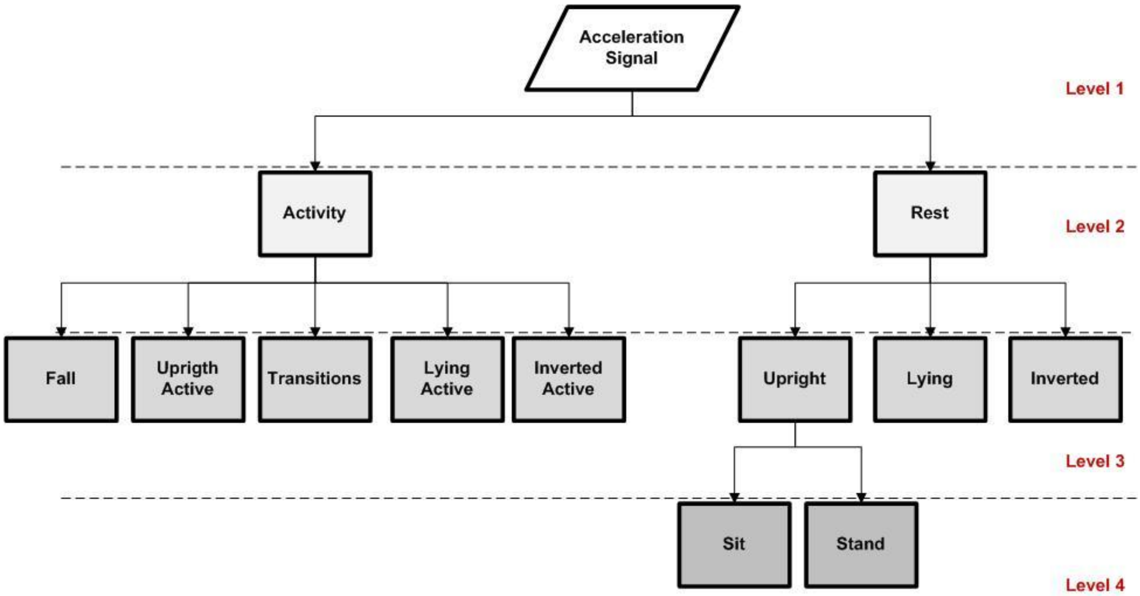


Figure 25- Hierarchical structure of the movements.

The upright active enclosed represents walking and climbing up/down stairs. The transitions include transitions between sit and stand postures and between sit and lying postures. The rest encloses tree main postures that are Upright, Lying and Inverted. In the fourth level the postures have a higher level of specificity. These postures are sit and stand, and they are originated by the upright posture from the third level.

7.1.3 Methods of decision

The acceleration signal from the TA accelerometer combines the body movements acceleration with the acceleration of gravity. These two components are separated and they will be used individually in different parts of the algorithm.

The first decision to take in relation to the algorithm is its time interval to process the data. The reason for this is because data processing has to be done in real time and it has to be processed in short time intervals. The ideal size of the intervals is between 0.8 and 1.4s (19). The time interval chosen to use in this project was 1s, since this value is within the ideal interval, which facilitates the implementation of the algorithm. Besides that, using a 1 second interval simplifies the calculations that use the sample frequency of the accelerometer (100Hz, i.e. one hundred values of acceleration from each axis per second).

The activity and rest periods are differentiated by a threshold value. This threshold is calculated through the signal variation in all three axes, which refers to the SMA. The SMA value is calculated through the equation (3). (19) (25)

$$SMA = \frac{1}{t}\left(\int_{o}^{t} |x(t)|dt + \int_{o}^{t} |y(t)|dt + \int_{o}^{t} |z(t)|dt\right) \quad (3)$$

The value chosen to the threshold is 0.89 g and it is calculated every second by summing all one hundred values progressively. This value is obtained through the BA component (26). In the following figure is possible to see the chart of the SMA value. The red line represents the SMA threshold value.

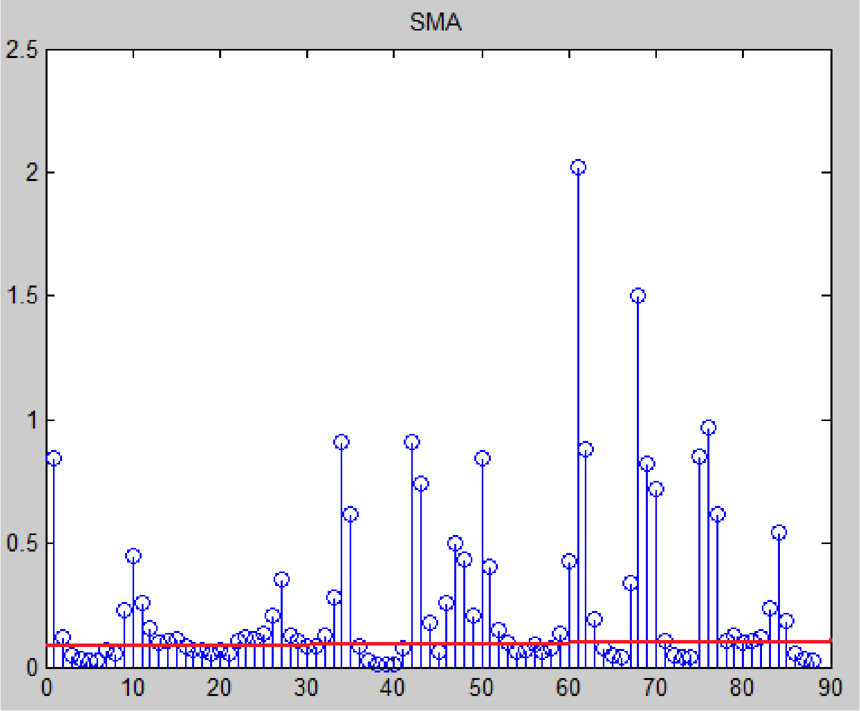


Figure 26- Representation of threshold value of SMA.

The differentiation method among the different postural orientations in moments of rest is based on the evaluation of the tilt angle (Φ), which defines the angle between positive Z-axis vector and the gravitational *g* vector. This method uses the gravitational component of the acceleration. The angle is calculated through the equation (4). (19)

$$\Phi = \arccos(z) \qquad (4)$$

The arc cosine is a limited function that may vary between the -1 and 1, so the Z acceleration values may also vary within this range. The output of the LPF, to which the function is applied, should also be limited between -1 and 1g.

In periods of rest the angles between 0º and 60º correspond to an upright posture, the angles between 60º and 120 º concern to a lying posture and the angles within the range of 120º to 180 º concern to inverted postures. The inverted posture is unusual, but it can happen after falling down the stairs. The upright posture is subdivided in two postures, sitting and standing, the transition's angle between these two postures may be configured in the Windows Application, for the reason that the sitting posture inclination may vary each person. Nevertheless, this angle's value is usually implicit as 10º.

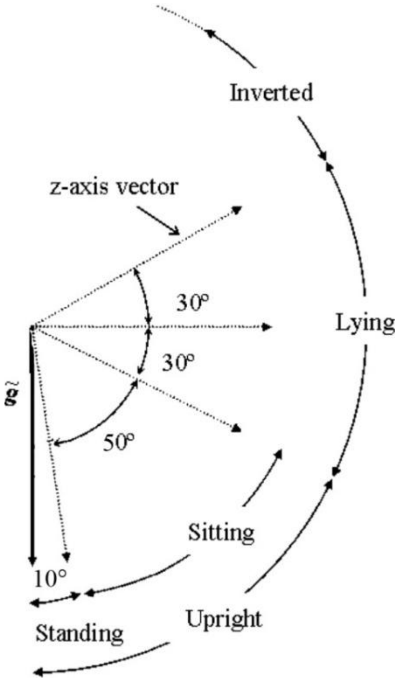


Figure 27- Method to determine the postural orientations using the tilt angle in periods of rest.

In periods of activity, four different postures are classified: upright active, transitions, lying active and inverted active, which can be seen in Figure 28. Periods of upright activity may include activities such as: walking, climbing up / down stairs and they are comprised between 0º and 10º. The transitions may vary between 10º and 60º. The angle between 60º and 120º corresponds to lying active periods and enclose the transitions between the different lying positions such as: lying front, lying back, lying right and lying left. The inverted active movements are the most unusual and may vary between 120º and 180º.

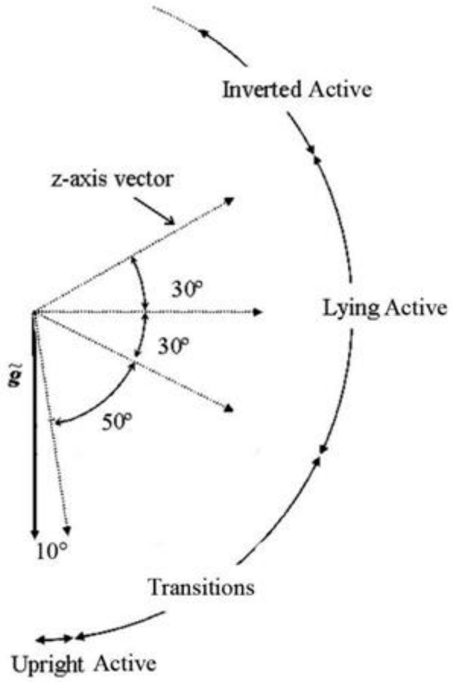


Figure 28- Method to determinate postural orientations in periods of activity, using tilt angle.

In this algorithm the movements is ordered considering the most critical movements first, in this case the falls.

The algorithm flowchart, taking into account all the decisions mentioned previously, is represent in Figure 29.

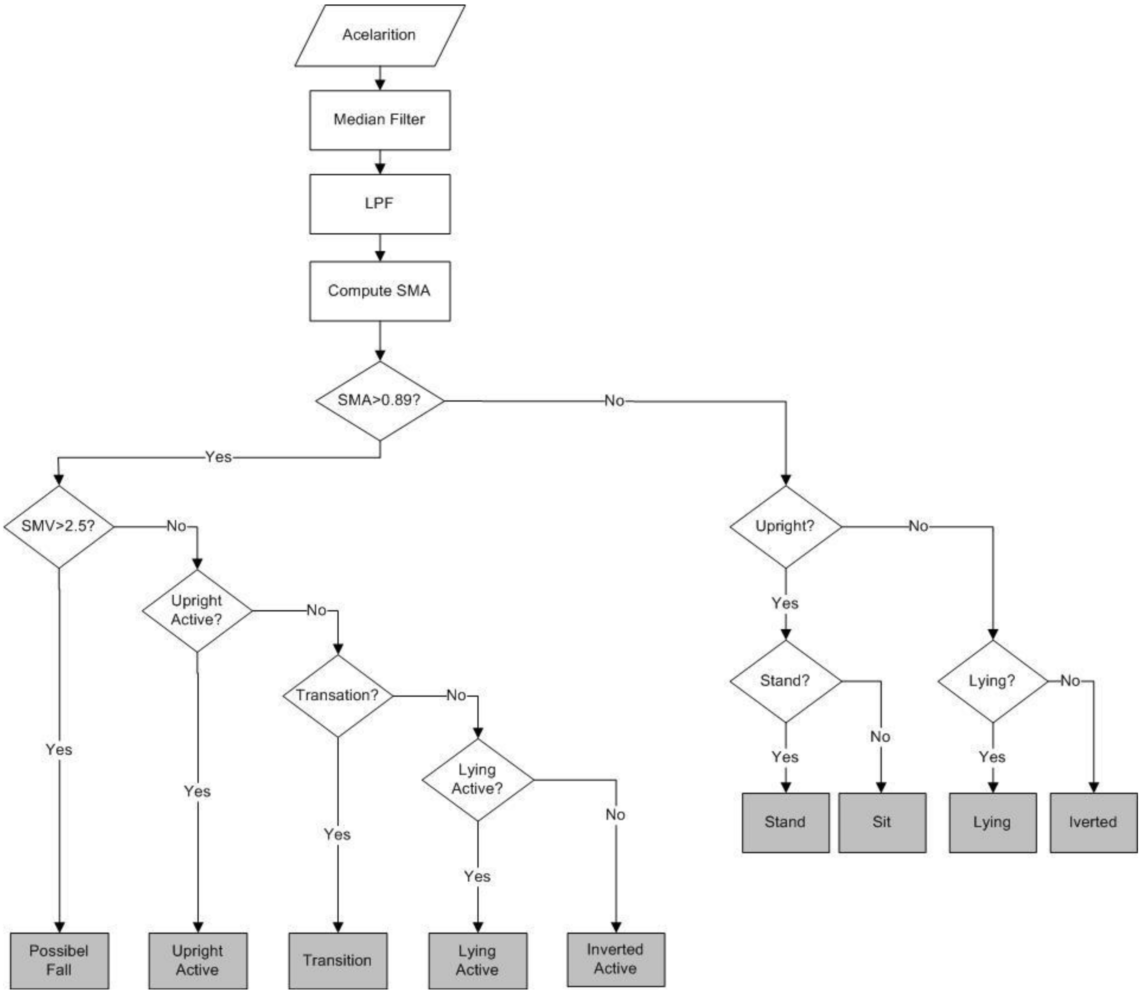


Figure 29- DRA Algorithm's flowchart.

7.2 Fall Detection Algorithm

To be considered as having a good performance, an algorithm for falls detection has to be able to detect several types of falls and distinguish them between stumbles and collisions. Furthermore it has to reduce the number of fake falls. The algorithm was developed taking into account these facts.

A possible fall is detected based on a threshold value, which is computed through the signal magnitude vector (SMV). The SMV value is computed through the equation (5). (19) (27)

$$SMV = \sqrt{x^2 + y^2 + z^2} \qquad (5)$$

This variable value measures the degree of movement intensity. A good threshold value should not be too high to detect falls, both in hard floor and in soft floor, or too low to avoid confusion between falls and DRA more abrupt. The threshold value selected was 2.5g (27). The SMV value is compute from the median filter output.

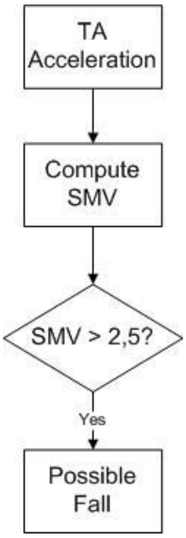


Figure 30- First decision of the detection fall algorithm.

When the value of SMV exceeds the threshold value a possible fall may have occurred. However this fact can be derived from other type of occurrences such as: stumbles, collisions or jumps. This differentiation can be done using the information about positions, after the threshold value has been exceeded (19) (28). This method assumes that after the fall the position is lying, for the reason that it is the most common situation. However, in several kinds of falls like falls from the chairs or falls that occur during climbing up or down stairs, that method doesn't work.

Since this is the method chosen for distinguishing between falls and other events that exceed the threshold value like stumbles and collisions, and it is also capable to identify different kinds of falls it is based on the information about the activity level after the threshold value has been exceeded. In other words, after the threshold value is exceeded the value of the SMA is saved for ten seconds, the five values for the five first seconds are ignored due the potential residual movement related to the fall (19). After 10 seconds, if the average of the SMA value is lower compared to the SMA threshold value, it means that, after the SVM threshold value has been exceed, the patient is immobilized and a fall may have happened.

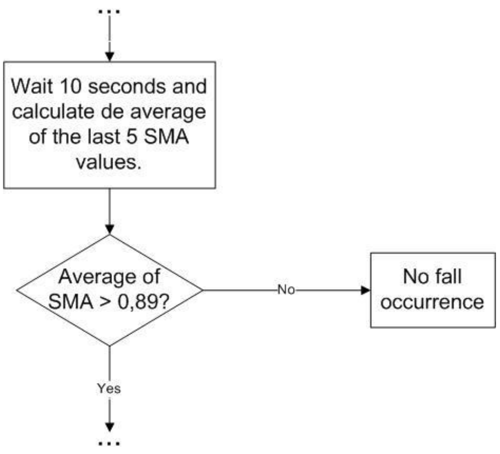


Figure 31- Second decision of the detection fall algorithm.

When this happens the Wiimote, which is placed on the waist at the bottom of the spine, starts to rumble .The choice of this place is related to the fact that the sensor must be near to center of mass of the body, to acquire a more accurate signal. In the Annex C, the correct way to place the Wiimote in patient's body is shown. When the Wiimote start to rumble, the patient has some time to click a button. This functionality avoids call for help when a fake fall occur, or when the patient falls but he is ok after it. The period of time that the Wiimote is rumbling and the button used to click, are two parameters configurable in the Windows Application.

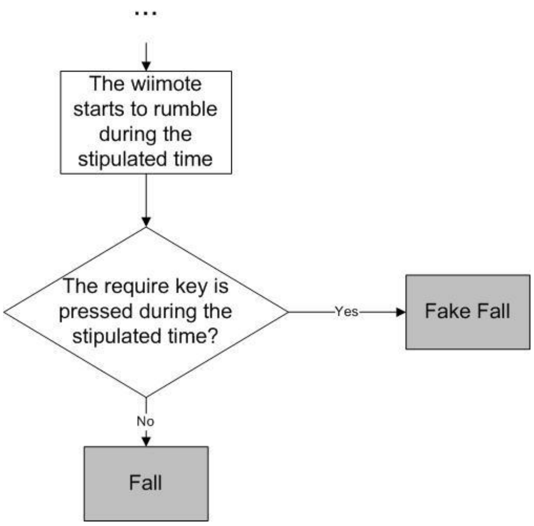


Figure 32- Last decision of the detection fall algorithm.

The complite fall detection algorithm is showed in the Figure 33.

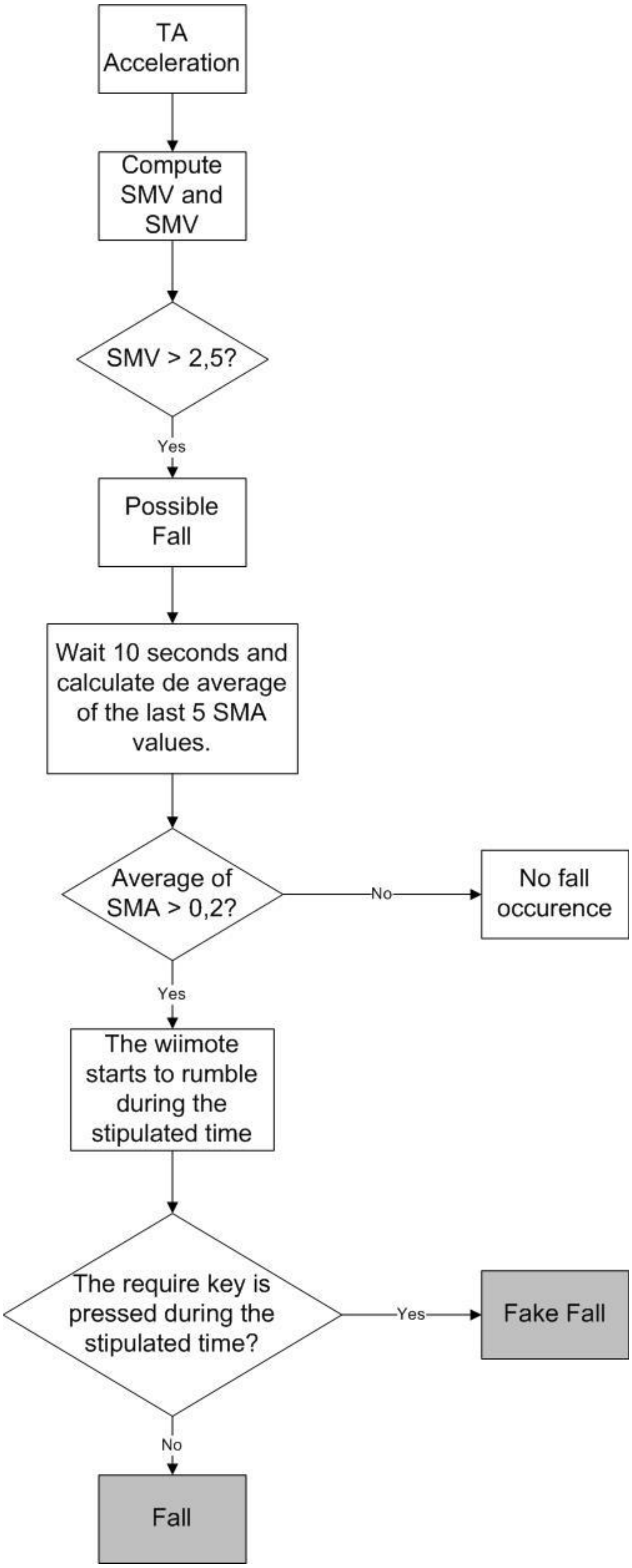


Figure 33- Fall detection algorithm’s flowchart.

7.3 Energy Expenditure Calculation

The Energy Expenditure is a parameter that can be calculated by using acceleration signals.

The TA accelerometer is used to evaluate the relationship between the energy expenditure during the physical activity due to the body acceleration and during sedentary activities and walking (29).

The estimation of energy expenditure is based on the equations (6) and (7):

$$IAA_{tot} = \int_{o}^{t} |x(t)|dt + \int_{o}^{t} |y(t)|dt + \int_{o}^{t} |z(t)|dt \qquad (6)$$

$$EE_{act} = 0.104 + 0.023\, IAA_{tot} \qquad (7)$$

The IAA_{tot} represents the integral of the absolute value of acceleration and is computed from the body acceleration component. The EE_{act} represents the energy expenditure in WKg^{-1}. As the energy expenditure due to sedentary activities and walking is quite different, they are studied separately. However this linear regression already encloses the energy expenditure due to sedentary activities and walking. To the home environment where the *AAL* Safe is supposed to work this estimative is adequate.

As the previous algorithm, this parameter is computed each second; due to this fact it can be concluded that IAA_{tot} is equal to SMA.

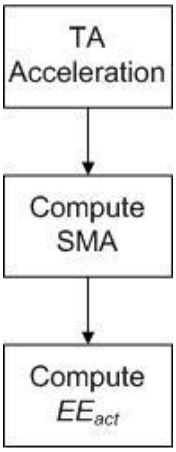


Figure 34- Energy expenditure's flowchart.

7.4 Integration of all Algorithms

In the figure bellow is showed all algorithms integrated. When a possible fall occur the part of the algorithm that check if a fall was real will work simultaneously with the part of algorithm responsible to distinguish between DRAs.

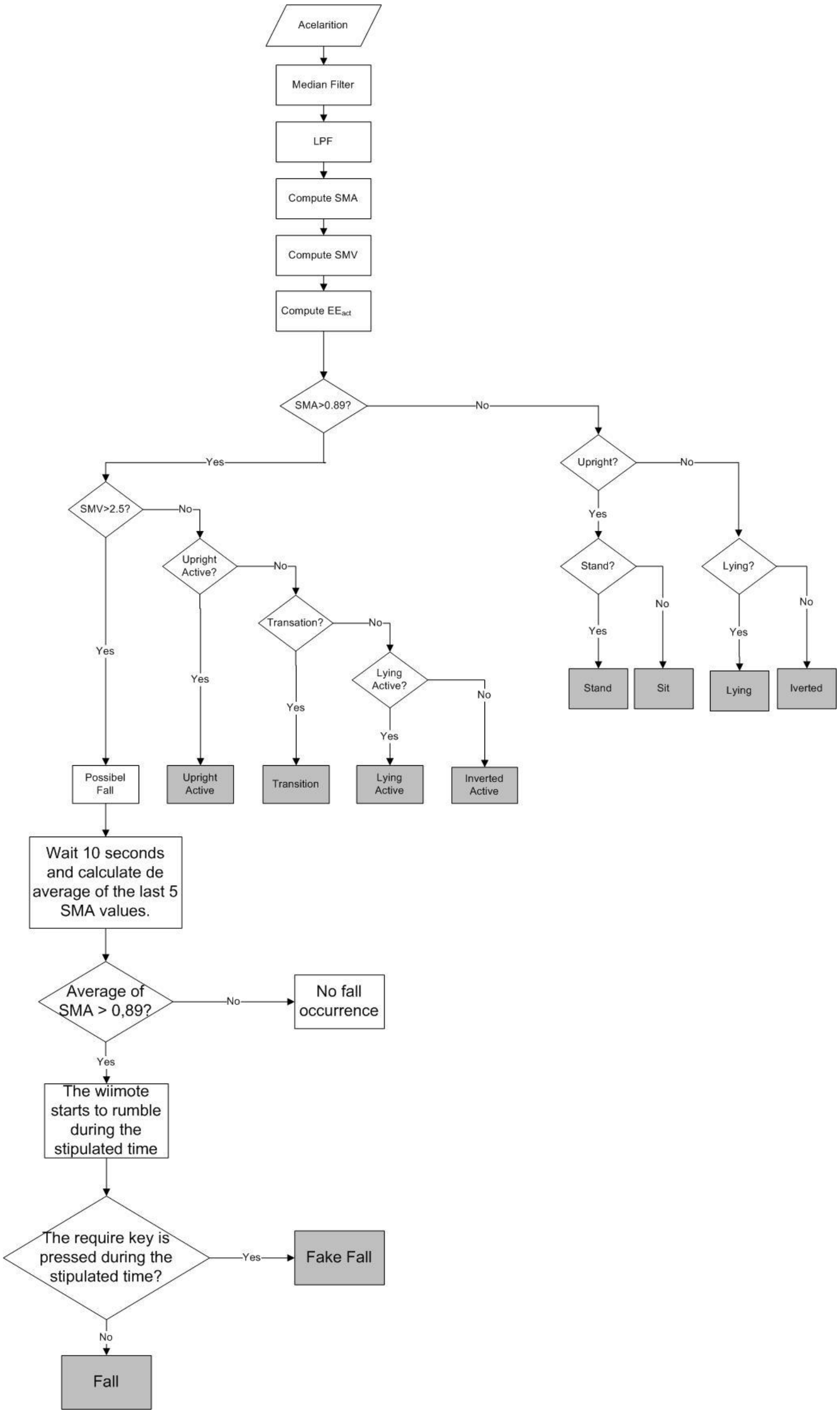


Figure 35- Flowchart with all algorithms integrated.

8 SOFTWARE DEVELOPMENT

In this Chapter will be described all processes about software development both in Matlab and in C#, such as the process of integration of this two programming languages.

The software development, in a first step begun with C# programming, but due to difficulties in signal processing, it was opted to test it and implement the algorithm previously in Matbab. Subsequently, when the algorithm was implemented in Matlab it was adapted to C# and integrated in the processing data module of the final system.

8.1 Matlab

The Matlab is an interactive system developed aiming at a simplification of mathematical calculation, which is based on matrix representation. Currently the Matlab represents a powerful development tool due to its high performance and its great diversity of toolboxes.

For this project, Matlab represented a great tool of research and development, all processing signal was done and tested in Matlab previously.

8.1.1 Matlab GUI

The Matlab GUI was developed to simplify the signal processing analysis. To be possible to test the acceleration data using the Matlab, first it has to be saved on a text file through the Wiimote Library in a C# program. The samples are saved simulating DRA and falls with the Wiimote attached to the person's body. While simulating DRAs and falls, as well as other kind of movements as stumbles and collisions, it is also being noted the time sequence of the events, and the moment that occurred. The samples usually have a duration of one to two minutes.

The samples saved in a text file from the C# program, are transformed in Matlab matrices using Spreadsheet Link, which is a supplement from the Matlad to Excel.

This process is repeated four times, one for each acceleration axis: X, Y and Z; and the other related to the SMV value, useful to the falls study. All of the four matrices are saved in .mat file and can be accessed from the Matlab GUI.

The Matlab GUI has a simple design. The GUI has a button to access the data (1), through which it is possible to select between the several samples with different sequence of events (2).

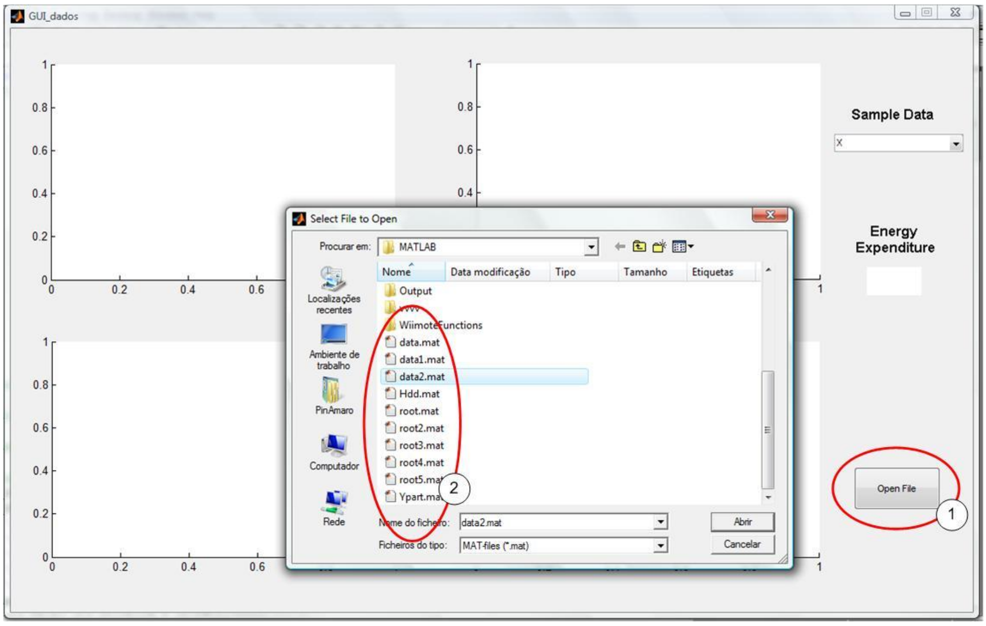


Figure 36- Overview of the Matlab GUI.

The GUI also has a pop-up menu (3), to choose between the acceleration from the three axes and SVM value. After selecting the sample data in the pop-up menu, automatically appears a label that shows the energy expenditure (4), and four graphics to enable the comparison between the initial signal and other different ways of processing.

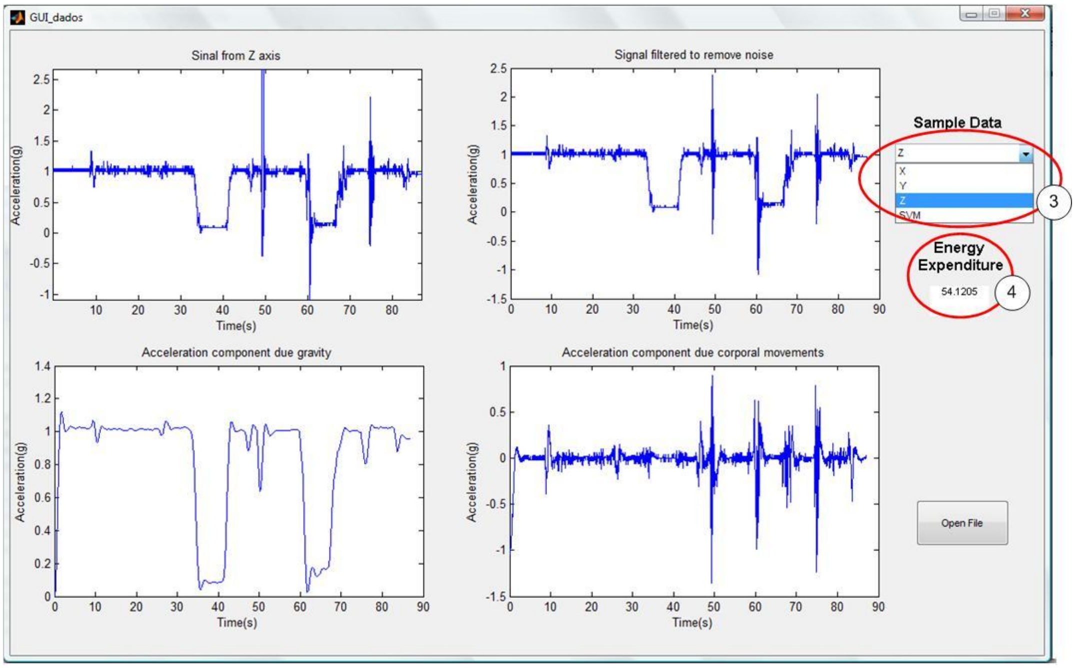


Figure 37- Overview of Matlab GUI after select the type sample data.

With the Matlab GUI it is possible to identify the best way of signal processing and to structure the algorithm. In this point of the project, after the way the signal processing is chosen and the algorithm is structured the possibility of interact the Wii directly with Matlab was tested. This hypothesis was tested using a Wii library from Matlab, but this interaction presents some limitations.

So it was decided to finish the implementation of the algorithm in C#, for the reason that the interaction between the Wiimote and C# is better and also for allowing the integration of the algorithm with the Windows Application.

8.1.2 Matlab tools used

The Matlab provides a set of tools and solutions, which facilitate the programming and make it more interactive and intuitive. The Matlab tools used in the development of this project were the Filer Design & Analysis Tool (FDATool) and the Deployment Tool (DeployTool).

8.1.2.1 FDATool

The FDATool is an interactive tool that allows the design analysis of all kinds of Finite Impulse Response (FIR) and IIR filters. Depending on the type of filter the parameters that characterize them are changed. After designing the desired filter, is possible to use it in several applications. It was through this tool that the LPF used was projected to process the acceleration's signal. In the Annex D is displayed an overview of the FDATool.

8.1.2.2 DeployTool

The Deploytool is another interactive tool that allows the conversion of Matlab functions in libraries to another programming language, easily and quickly.

On Matlab the code with the desired functions is created and saved in an .m file. On DeployTool is selected the "Matlab Buider NE" option and next the ".Net Componet" option. These options are specific for .NET applications, which is the case of C# VisualStudio. After the build component a .dll file is generated, which will be added to the C# application. In the Annex E can be seen an overview of the DeployTool.

To interact the Matlab library created with DeployTool in C# application it is necessary one more library, the MWArray.dll, contained in a Matlab folder. This library is always needed to do the interaction.

8.2 C#

8.2.1 Interaction between Matlab and C#

Though the Matlab was essential to the Project development, it would not be complete, and wouldn't make sense if the algorithm was not implemented in real time.

The final algorithm is inserted on the processing module of the system, corresponding to the monitorization area of the Windows Application.

The first challenge in implementing the algorithm in C# was finding a way to process the signal. This problem was solved, using the DeployTool from the Matlab.

Through the DeployTool, two libraries were created one of them with the function to compute the median filter and another to compute the LPF.

The created libraries and their respective functions and classes are shown in the Table 13.

Table 13- Classes and functions of the Libraries created on DeployTool.

	Library to compute Median filter	**Library to compute LPF**
Name of libraries	IIRfil.dll	MedianFilter.dll
Classes of the libraries	IIRfilclass	MedianFilterClass
Functions	IIRfiltro	Medfilter

The names of the functions correspond of the names of the .m files created on Matlab.

The implementation of this kind of libraries created on Matlab, is done through MWArray library, which has the function to do the interaction between the Matlab and C#.

Table 14- MWArray's used calss

Name	MWArray.dll
Used class	MWNumericArray

The algorithm implementation starts with the acquisition of the acceleration, from the 3 axes, which are saved in lists, with a frequency of 100 values per second. The time control of the algorithm is done by a Timer, for each second elapsed 100 samples are analyzed and a decision is made.

After the samples are acquired, the two filters are applied. This process can be consulted in Annex F.

8.2.2 Limitations of the LPF

One of the LPF's filter limitations is its stabilization time. From what was observed in Figure 23, that time is about the 1.5s, that means 150 sample. As the algorithm is processed in windows of 1s, it is not possible to get the correct values of the filter application in so small windows. To solve the problem something had to be done, 3s of samples were acquired, the data of the first 2s were eliminated and the decision was taken after the 100 values referring to the last second.

In practice only 2s of monitoring one lost. For example on the 4th monitoring second the values of the 2nd and 3rd seconds are used to calculate the filter and in the end to take a decision only the value of the 4th second is used. In this way it is possible to filter the signal allowing the filter stabilization without loosing any information data.

The other limitation is related to the fact the LPF output overcomes the expected values. These values can only be overcome in the standing or inverted position, which means the two extremities. To solve this problem and making the calculation of the Equation 4 always possible it was assumed that when the values are superior to 1 or inferior to -1 they are considered equal to them.

8.2.3 Algorithm decisions

The algorithm decisions are always done based in the matrix values. Firstly the SMA is calculated value to verify if the patient was in a period of activity or in a period of rest. Later the position is identified through the average calculation of the 100 values of the patient's tilt angle.

The final goal of the algorithm is count the time that each patient dispends in each position.

Table 15- Algorithm's decisions.

Decisions of the algorithm	SMA value	Tilt Angle variation	Time category	Visual information available in the application
Standing	<0.89	0º a 10º	Standing Time	🟢 **Stand**
Sitting	<0.89	10º a 60º	Sitting time	🔵 **Sit**
Lying	<0.89	60º a 120º	Laying time	🔴 **Lay**
Inverted	<0.89	120º a 180º	Undefined position time	Unavailable information
Possible fall	>0.89	SMV > 2.5	Undefined position time	Unavailable information
Inverted active	>0.89	120º a 180º	Undefined position time	Unavailable information
Upright active	>0.89	0º a 10º	Standing Time	🟢 **Stand**
Transitions	>0.89	10º a 60º	Transitions Time	Unavailable information
Lying active	>0.89	60º a 120º	Lying Time	🔴 **Lay**

As it is possible to see on the table above, each position is related to a kind of time. There are five categories, standing time, sitting time, Lying time, time of transitions and undefined position time.

For each second passed a second in one of these 5 categories is incremented.

When the SMV value surpasses the 2.5g a seconds in the category of the undefined position time is counted. Moreover when the SMV threshold is exceeded it is registered if a fall occurred or if it was a fake fall. In the case of the SMV threshold value surpasses the 2.5g and a fall or a fake fall doesn't happen, it is because a more abrupt movement happen, like a jump, a stumble or collision.

When the algorithm detects what seems to be a fall, the fall algorithm will operate independently from the other one checking if the fall was real.

Simultaneously, the DRA algorithm continues to run and saving the daily activities of the patient.

When a fall occurs a beep is activated and visual alert appears in the Windows Application.

The energy expenditure is calculated in each second, from the SMA value and incremented. The total value of the energy expenditure is showed at the end of the monitorization.

When the monitorization ends a table with the statistics concerning the monitorization is shown, the percentage of the elapsed time, the number of falls and fake falls, the energy expenditure and the date of the beginning and end of the monitorization.

9 TESTS

To test the algorithm performance several tests were done. The tests were made by the student during and after the implementation of the algorithm in C#. To test the algorithm the Wiimote has to be connected to the application, which has to be able to receive the data from the Wiimote.

The tests are shown in Table 18, they are identified by an ID (TST+Identification+number of the test), as the test was done to the algorithm the identification is an “A”. Each test has a description related to the intention of the test, and has an expected behaviour. If the expected behaviour corresponds to real behaviour the success of the test is positive and it is marked with ✓, if the real behaviour doesn’t correspond to the expected behaviour the test is marked by a ✗.

Table 16- Algorithm's tests.

Test ID	Test	Expected Behaviour	S
TSTA01	Identify Lying Position	Appear description "Lying" in application and the time is incremented to "Lying time"	✓
TSTA02	Identify Stand Position	Appear description "Stand" in application and the time is incremented to "Standing time"	✓
TSTA03	Identify Sit Position	Appear description "Lying" in application and the time is incremented to "Sitting time"	✓
TSTA04	Identify inverted position	The time is incremented to "undefined movements time"	✓
TSTA05	Identify Lying active	Appear description "Lying" in application and the time is incremented to "Lying time"	✓
TSTA06	Identify Transitions	The time is incremented to " Transitions time"	✗
TSTA07	Identify Upright active	Appear description "Stand" in application and the time is incremented to "Standing time"	✓
TSTA08	Identify Inverted Active	The time is incremented to "undefined movements time"	✓
TSTA09	Identify Possible fall	The time is incremented to "undefined movements time", and is computed the average of SMA value.	✓
TSTA10	Identify fake fall	The stipulated key is pressed during the stipulated time and the number of fake falls is incremented	✓
TSTA11	Identify fall	The number of fall is incremented	✓

10 CONCLUSION

The AAL Safe was developed with the objective of thinking about the elderly and their needs and limitations, improving their security in the care institution environment. The *AAL* Safe can alert when a fall occurs and call for help almost instantly. It can also do a constant monitorization of the patients' activities and energy expenditure. These two last parameters allow an evaluation, by caregivers, about the level of the activities of the patient that to allow to conclude if the patient has an active life or a sedentary life.

All objectives of the project were achieved, it comprehends a Solution capable to monitor in real-time the acceleration signal from the Wiimote attached of the patient's body, and through this can taking decision related to the posture of the body, detecting a fall accurately and measuring the energy expenditure.

Concluding, and taking into account the limitations of the project regarding the Wiimote design and its game functions, the final result surpasses the student's expectation.

10.1 Future Work

Regarding to the future work, several aspects can be improved. The replacing of the Wiimote by a more wearable and appropriate device, is the most important work to develop in the near future. It will be interesting that the final device, not only includes an accelerometer, but also incorporate a gyroscope. These two sensors associated will make possible an accurate detection of movements. This possibility will be tested with the Wii MotionPlus, before being developed.

Concerning the signal processing several things may be done. It will be important to develop a filter that presents better results, like a smaller stabilization time, and a filter than doesn't surpass the expected limits of the GA component, allowing an easier implementation. It will also be interesting to detect specific activities like walking and climbing up/down stairs and transitions between sit and stand, and between stand and lying. To detect these activities and transitions the processing signal implemented is not enough, It will be necessary to test new ways to process the signal like wavelets and FFT, which represent better results in what concerns the detection of specific movements.

To obtain more accurate results about the algorithm performance, the tests should be done by a significant sample of persons, of an age group near to the persons that will use the system and with a range of weights and heights. The tests should done in periods of long duration simulating all the activities that the system intends to recognize.

10.2 Final Appreciation

The AAL represents currently a major research and development program that has the endeavor of solving the problems related with the ageing society. So, bearing this in mind and in order to contribute to the improvement of the elderly lifestyle, this project was created and developed. At a personal level, this challenge has represented a fulfillment for the student.

The development of the project itself was somehow imposed, since it did not correspond to his initial expectations/choice. However, as the time has gone by and with the decision of developing a fall detector, the student has become more motivated and committed to his work.

At a professional level, the project has revealed itself as valuable since the student had the chance to contact with a business environment and earn some experience for future works. In addition, the acquired knowledge was important since the undergraduate could acquire knowledge in areas of great importance and little explored in the course, as the programming and signal processing is.

References

1. *Ambient Assisted Living.* [Online] [Cited: 2009 July 2009.] http://www.aal-europe.eu/about-aal..

2. *Ambient Assisted Living.* [Online] [Cited: 15 July 2009.] http://www.aal-europe.eu/.

3. *Ambient Assisted Living.* [Online] [Cited: 2009 July 16.] http://ec.europa.eu/information_society/activities/einclusion/docs/ageing/ageing_launch_fdb.pdf..

4. [Online] [Cited: 2009 July 16.] http://siteresources.worldbank.org/SOCIALPROTECTION/Resources/SP-Discussion-papers/Labor-Market-DP/0703.pdf..

5. *Instituto Nacional de Estatistica.* [Online] [Cited: 2009 July 14.] http://www.ine.pt/xportal/xmain?xpid=INE&xpgid=ine_publicacoes&PUBLICACOESpub_boui=65944632&PUBLICACOESmodo=2..

6. *Ambient Assisted Living.* [Online] [Cited: 12 July 2000.] http://ec.europa.eu/information_society/activities/einclusion/docs/ageing/staff_working_paper_ageing.pdf..

7. Alert One. [Online] http://www.falldetection.com/iLifeFDS.asp.

8. [Online] http://www.tunstall.co.uk/assets/literature/6_1_41falls_management_solutions_sheet.pdf.

9. [Online] http://www.atdementia.org.uk/productDetails.asp?searchmethod=cats&cat_id=9&subcat_id=13&product_id=89&page=1&referer=productSearch[d]asp[q]search[e]go[a]searchmethod[e]cats[a]cat_id[e]9[a]subcat_id[e]13[a]page[e]1.

10. *HaloMonitoring.* [Online] http://www.halomonitoring.com/halo/Caregivers/myHaloProductOverview/tabid/119/Default.aspx.

11. **Wronski, Michal Piotr.** *Design and Implementation of a Hand Tracking Interface using the Nintendo Wii Remote.* Cape Town : s.n., 2008.

12. Engadget. [Online] [Cited: 12 July 2009.] http://www.engadget.com/2009/06/02/nintendo-wii-vitality-sensor-detects-your-pulse/..

13. [Online] [Cited: 20 July 2009.] http://en.wikipedia.org/wiki/Wii_MotionPlus.

14. **Pires, J. Norberto.** *Report on state of art of interaction devices and mechanisms* . 2005.

15. **Pedro Neto, J. Norberto Pires, Member,IEEE.** High-Level Programming for Industrial Robotics: using Gestures, Speech and Force Control.

16. **Parkway, Alton.** *BCM2042 Product Brief*. 2006.

17. *Wiibrew.* [Online] [Cited: 22 July 2009.] http://wiibrew.org..

18. **DEVICES, ANALOG.** *Small, Low Power, 3-Axis ±3 g iMEMS® Accelerometer*. 2007.

19. **Matos, Tomé.** *Remote Vital Signs Monotiring- Instrumentation Module*. 2008.

20. Bluetooth. [Online] [Cited: 27 July 2009.] http://bluetooth.com/Bluetooth/.

21. *Bluetooth Protocol.* [Online] [Cited: 27 July 2009.] http://www.baracoda.com/shared_docs/bluetooth_protocol.pdf..

22. *Coding4Fun.* [Online] [Cited: 28 July 2009.] http://blogs.msdn.com/coding4fun/archive/2007/03/14/1879033.aspx..

23. **Stephen J. Preece, Jonh Yannis Goulermas, Member, IEEE, Laurence P. J. Kenney, and David Howard.** A Comparison of Feature Extraction Methods for the Classification of Dynamic Activities From Accelerometer Data. 2008.

24. **Dean M. Karantonis, Michael R. Narayanan, Merryn Mathie, nigel H. Lovell, Branco G. Celller.** Implementation of a Real-Time Human Movement Classifier Using a Triaxial Accelerometer for Ambulatory Monitoring. 2005.

25. [Online] [Cited: 15 August 2009.] http://books.google.com.br/books?id=0Ki_6cjv7W0C&pg=PA63&dq=median+filter+definition#v=onepage&q=median%20filter%20definition&f=false.

26. [Online] [Cited: 15 August 2009.] http://www.cs.tau.ac.il/~turkel/notes/meanmed.pdf.

27. [Online] [Cited: 18 August 2009.] http://images.google.pt/imgres?imgurl=http://www.its.bldrdoc.gov/projects/devglossary/images/lpfiltec.gif&imgrefurl=http://www.its.bldrdoc.gov/projects/devglossary/_low-pass_filter.html&usg=__XXx7vrwW_bYuTlAxR-9Ra12tuXE=&h=328&w=382&sz=5&hl=pt-BR&start=1&.

28. [book auth.] John L. Semmlow. *Biosignal and Biomedical Image Processing MATLAB based applications.*

29. **Fahrenberg, Friedrich Foerster and Jochen.** Motion pattern and posture: Correctly assesed by calibrated accelerometers. 2000.

30. **Mathie, M. J., et al.** Classification of basic daily movements using a triaxial accelerometer. 2004.

31. **Mathie, M. J., et al.** Detection of daily physical activities using a triaxial accelerometer. 2003.

32. **Calijn V. C., Karel T. M. Koekkoek, Maarten verduin, Rens Kodde, and Jan D. Janssen.** *A Triaxial Accelerometer and Portable Data Processing Unit for the Assessment of Daily Physical Activity.* 1997.

33. **M.J.Mathie, J.Basilakis, B.G.Celler.** A System for Monitoring Posture and Physical Activity using Accelerometers . 2001.

34. **Klaus-Hendrik Wolf, Arne Lohse, Michael Marschollek, Reinhold Haux.** Development of a Fall Detector and Classifier based on a Triaxial Accelerometer Demo Board. 2007.

35. **Carlijn V. Bouten, Klaas R. Westerterp, Maarten Verduin, Jan D. Janssen.** Assessment of energy expenditure for physical activity using a triaxial accelerometer. 1994.

Annexes

Annex A -Table of the HID descriptor blocks for all communication between the Wiimote and the host.

I/O	Report ID	Payload Size	Function
O	0x10	1	Unknown
O	0x11	1	Player LEDs
O	0x12	2	Force Feedback
O	0x13	1	IR Sensor Enable
O	0x14	1	Enable Speaker
O	0x15	1	Controller Status
O	0x16	31	Write data
O	0x17	6	Read data
O	0x18	21	Speaker data
O	0x19	1	Mute Speaker
O	0x1a	1	IR Sensor Enable 2
I	0x20	6	Status Information
I	0x21	21	Read Memory and register data
I	0x22	4	Acknowledge output report
I	0x30-0x3f	2-21	Data Reports

Annex B- WiimoteLib Using

To work with WiimoteLib, the first step is to add a reference to the WiimoteLib.dll in the desired application. With the WiimoteLib.dll added, all functionalities of the library are available and it is already possible to instance the Wiimote class and start using it. The Wiimote class allows to setup events, setup the report type of the data that will be returned, and call some methods as Connect and Disconnect methods.

The library has a class entitled WiimoteCollection that allows using multiple Wiimotes through the FindAllWiimotes method. Thus it is possible to use each individual Wiimote object of the collection as a separate instance.

The data can be acquired from the API in two ways. The first way is through event mode; in this case the WiimoteChanged event must be subscribed. The second way is simpler; it just needs retrieve information from the WiimoteState property of the Wiimote class.

There are several types of reports, all of them capable to produce the required data for the Wiimote and all current extensions. These reports can be accessed through the SetReportType method, of the Wiimote class (18). The reports are:

- **Buttons** - Button data only
- **ButtonsAccel** - Button and accelerometer data
- **IRAccel** - Button, accelerometer and IR data
- **ButtonsExtension** – Button and extension data
- **ExtensionAccel** - Button, accelerometer and extension data
- **IRExtensionAccel** - Button, accelerometer, extension and IR data

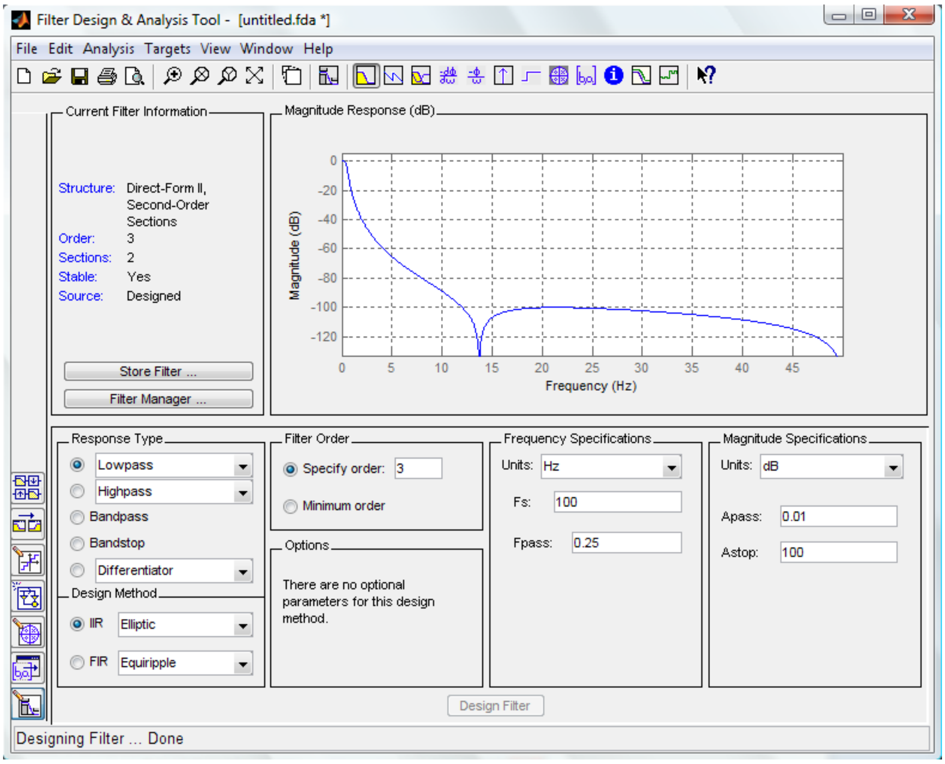
In the following table the most important commands of the Wiimote API and their respective functionalities are presented.

Command	Functionality
Wiimote wm = new Wiimote();	Create a new instance of the Wiimote class
wm.Connect();	Connect to Wiimote
WiimoteCollection wc = new WiimoteCollection(); wc.FindAllWiimotes()	Find all Wiimotes connected via BT
wm.SetReportType(Wiimote.InputReport.IRExtensionAccel, true);	Set the report type to return all data from the Wiimote

Void wm_WiimoteChanged(object sender, WiimoteChangedEventArgs args)
{
WiimoteState ws = args.WiimoteState;
Debug.WriteLine(ws.ButtonState.A);
}
	Return the state of the button A

Annex C- Correct way to place the Wiimote in patient's body

Annex D- Overview of the FDATool



Structure: Direct-Form II, Second-Order Sections
Order: 3
Sections: 2
Stable: Yes
Source: Designed

Response Type:
- ● Lowpass
- ○ Highpass
- ○ Bandpass
- ○ Bandstop
- ○ Differentiator

Design Method:
- ● IIR Elliptic
- ○ FIR Equiripple

Filter Order:
- ● Specify order: 3
- ○ Minimum order

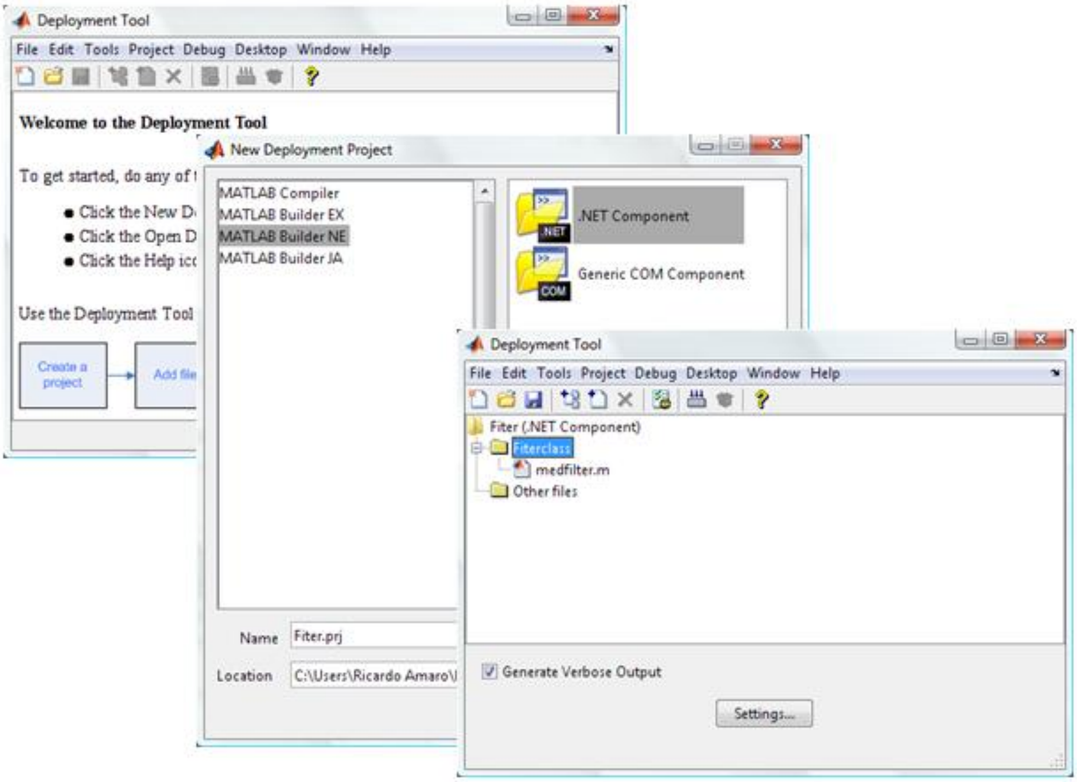
Options: There are no optional parameters for this design method.

Frequency Specifications: Units: Hz; Fs: 100; Fpass: 0.25

Magnitude Specifications: Units: dB; Apass: 0.01; Astop: 100

Designing Filter ... Done

Annex E- Overview of the DeployTool



Welcome to the Deployment Tool

To get started, do any of t

- Click the New D
- Click the Open D
- Click the Help ic

Use the Deployment Tool

Name: Fiter.prj

Location: C:\Users\Ricardo Amaro\

Fiter (.NET Component)
- Fiterclass
  - medfilter.m
- Other files

☑ Generate Verbose Output

Annex F- Matlab libraries used- commands and functionalities

Command	Functionality
IIRfilclass filtro1 = new IIRfilclass(); MedianFilterClass filtro = new MedianFilterClass();	Instantiate the classes IIRfilclass and MedianFilterClass
MWNumericArray _Z = new MWNumericArray();	Instantiate the class MWNumericArray creating a _Z matrix
_Z = m_DataZ.ToArray();	Converts the m_DataZ list values where they are saved, to a matrix created previously.
saida_Z = (MWNumericArray) filtro.medfilter (_Z);	Applies the first filter to the _Z matrix creating a new matrix called saída_Z, com the filtered signal.
saida_Z2 = (MWNumericArray) filtro1.IIRfiltro(saida_Z);	Applies the second filter to the saída _Z matriz creating a new matrix called saida _Z2, with the filtered signal