

Heart Failure monitoring system based on Wearable and Information Technologies

E. Villalba, M.T. Arredondo, M. Ottaviano, D. Salvi, E. Hoyo-Barbolla
Life Supporting Technologies, Technical University of Madrid, Spain
Email: {evmora, mta, mottaviano, dsalvi, evahb}@lst.tfo.upm.es

S. Guillen
Health and Wellbeing Technologies R&D, ITACA Institute, Valencia, Spain
Email: sguillen@itaca.upv.es

Abstract—In Europe, Cardiovascular Diseases (CVD) are the leading source of death, causing 45% of all deceases. Besides, Heart Failure, the paradigm of CVD, mainly affects people older than 65. In the current aging society, the European MyHeart Project was created, whose mission is to empower citizens to fight CVD by leading a preventive lifestyle and being able to be diagnosed at an early stage. This paper presents the development of a Heart Failure Management System, based on daily monitoring of Vital Body Signals, with wearable and mobile technologies, for the continuous assessment of this chronic disease. The System makes use of the latest technologies for monitoring heart condition, both with wearable garments (e.g. for measuring ECG and Respiration); and portable devices (such as Weight Scale and Blood Pressure Cuff) both with Bluetooth capabilities.

Index Terms—wearable systems, Bluetooth sensors, web services, health monitoring, personalized applications

I. INTRODUCTION

Heart Failure (HF) is a relatively common chronic disorder and is considered the paradigm of cardiac chronic diseases. This disorder mainly affects people older than 65 [1]. The increase on the life expectancy in the developed countries has resulted in an increase in the number of hospitalizations due to chronic diseases, as well as in a potential decrease in the quality of life of the aging population.

The Heart Failure Management System (HFMS) makes use of the latest technologies to monitor heart condition, both with wearable garments (to measure ECG and Respiration); and portable devices (such as Weight Scale and Blood Pressure Cuff) with Bluetooth capabilities [2].

HFMS aims at decreasing the mortality and morbidity of the HF population. The system also focuses on improving the efficiency of the healthcare resources, maximizing the cost-benefit rate of the heart failure management.

The main users of the system are: a) HF chronic disease management service provider, with cardiologists and nurses; and b) patients with HF.

HF Management System consists of three main elements: a) The Front-end, b) the User Interaction System and c) the Back-end. The following figure (Fig. 1) sketches the global system:

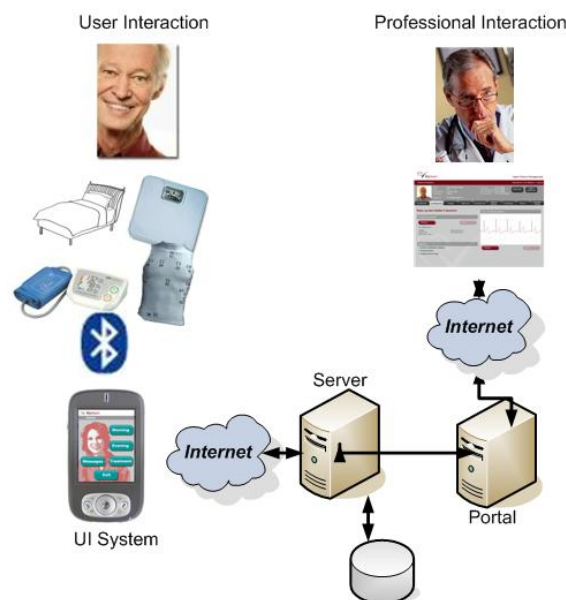


Figure 1. HFMS overview

The **Front-end** is composed of the different textile sensors and electronics to record the vital signals required by the application (ECG, Respiration and Activity).

Based on "A new solution for a Heart Failure Monitoring System based on Wearable and Information Technologies", by E. Villalba, MT Arredondo, S. Guillen and E. Hoyo Barbolla, which appeared in the Proceedings of the IEEE International Conference on BSN 2006, Cambridge, Massachusetts, US, April 2006. © 2006 IEEE.

Besides, some portable devices such as weight scale and blood pressure cuff are also used.

The **User Interaction System (UIS)**, which is based on a personal digital assistant (PDA) device which receives data from the monitoring devices, processes it and encourages patients in the daily care of their heart. Moreover, it enables the communication and synchronization with the Back-end.

The **Back-end**, which includes the processing server and the databases, manages the data gathered for every patient. Professionals can visualize and manage all data through a web access provided by a portal, which is based on Cocoon Framework [3]. The system collects personal and clinical data. Therefore, security issues play a role of major importance.

All the daily routine data are processed and used in the detection of functional capacity, symptoms worsening and other complications. The timing tendency of data is automatically assessed in order to enable an early detection of: a) possible clinical decompensations (clinical destabilization warning signs), b) continuous "out of hospital" arrhythmia risk stratification and, c) evaluation of the HF progression. On the other hand, motivation strategies were taken into account in order to provide patients with pertinent and relevant information, according to their physical and psychological status.

II. METHODS

The methodology applied is based on Goal-Directed Design [4], whose process can be divided into five iterative phases: Research, Modeling, Requirement, Framework and Refinement, as shown in the following figure:

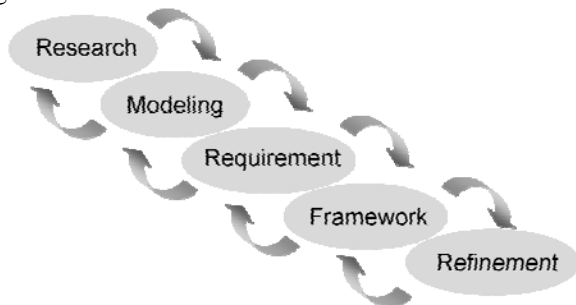


Figure 2. Goal-Oriented Design Phases

A. Research Phase

During the **Research** phase, interviews to HF population, medical specialists (cardiologist and nurses) and business managers related to the chronic disease area of hospitals, were carried out. Within these interviews a mock-up system was thoroughly validated during three months [5]. The validation was based on open and close-

ended questions followed by a system demonstration in order to allow the users to assess the usability and comfort of the system. In total 26 people were interviewed: 10 end users (9 men and 1 woman, 80% of them above 60 years), 6 business managers and 10 cardiologists. These interviews were designed by Philips Design Eindhoven in the remit of the MyHeart Project.

Each interview consisted of 5 sections: introduction, storyboard, tangible, conclusion and closure. During the introduction, the interviewer explained the purpose of the interview, offering the interviewee a confidential agreement to sign. Some general questions about the position towards technology and health status were posed at that stage.

Within the storyboard section the interviewer presented the global system and asked the interviewees open questions about their impressions and doubts. Once the basics of the system were grasped, the tangible section started, in which wearable garments, electronics, different user devices and the portal was presented. The user was also welcome to play with the different application early prototypes developed within a PDA. Thus, the users were able to determine whether the PDA was suitable for them, or they preferred other interaction devices such as smart phones and TVs.

During the conclusion section, the interviewer asked for the overall impression. In addition, the interviewee filled up a scoring sheet with 10 closed questions rated from 1 to 5, which provides a quantitative result. In the closure section, the interviewer thanked the participant and took some last notes about the experiences during the whole interview.

The time for each interview varied from 60 minutes for the professionals to about 120 minutes for the end users. The next table shows the interview process and timetable.

TABLE I. INTERVIEW FOR FINAL USERS

Section	Subject	Time
Introduction	Setting the scene – 5 min	15 min
	General questions – 10 min	
Storyboard	Reading – 10 min	25 min
	Questionnaire – 15 min	
Tangible	Garment aspects – 15 min	40 – 50 min
	Demonstrator and questions – 25 min	
Conclusion	Questions – 28 min	30 min
	Scoring – 2 min	
Closure		5 – 10 min
Total time		115–130 min

The following list displays an example of the scoring questionnaire for the final user.

1. I like this concept
2. I will use this concept
3. This concept will reduce my quality of life
4. This concept will motivate me to a healthier lifestyle
5. This concept will make me feel neglected
6. This concept will be a burden to my lifestyle
7. This concept will help me stay in control of my health
8. I will never trust this concept to look after my health
9. This concept will invade my privacy
10. This concept will offer me a pleasant experience

The final users answered these questions from 1 (totally disagree) to 5 (totally agree). The next figure shows the quantitative results out of the interviews.

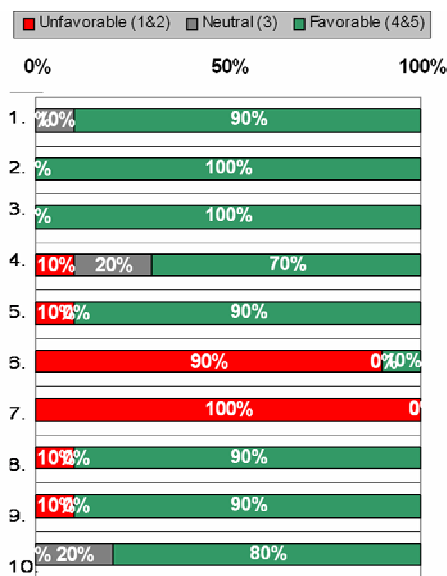


Figure 3. Final user scoring sheet

The overall attitude towards the system was very positive. Most interviewees found it useful and a good solution for the long term treatment of chronic disease.

The system provides a sense of security and confidence in people with HF, an increase in the quality of life that many users would be happy to pay. Also, this system is perceived to result in an increase of productivity in the Healthcare System. It allows the management of a larger number of people. Besides, this system motivates HF population to do moderated controlled exercise, which improves their quality of life and makes them conscious of the importance of their healthcare.

Nevertheless some issues were addressed as weakness. For instance, the system needs the user interaction with a technical device, which reduces the number of people that could be incorporated to the follow up of the program. Furthermore, physicians have a low level of trust on delivered signal quality.

The system could create in users high expectations such as 24 hours attention from physicians, which is not the aim of such a system.

The actual design is not appropriate to hot weather conditions. Moreover, some users have certain hindrance to tight clothes.

The system could be designed to incorporate a higher modularity, being able to offer different services to a diverse range of users, in function of their necessity.

B. Modeling Phase

Once the Research phase was completed, the **Modeling** phase generated both domain and user models, taking as input the results from the previous phase. Domain models included workflow diagrams. User models, or personas, are user archetypes that represent behavior patterns, goals and motivations.

The persona of HFMS is elderly people aged 65 or more. They are aware of their heart condition and are proactive to take a better care. Furthermore, they are able to handle an electronic device, following a very intuitive system. Besides, they do not have any special need in terms of accessibility (e.g. blind people).

Users should be able to easily start/stop the application, view their daily tasks, be alerted of the previous uncompleted tasks, follow instructions to perform monitored sessions, view results, answer questionnaires (e.g. 5 questions in the assessment of his mood), and consult messages from professionals.

C. Requirements Phase

The **Requirements** phase employed scenario-based design methods. End users were prompted to follow a daily routine divided into morning, exercise and before sleeping contexts.

A context is a set of tasks (also named activities) to be performed together by the user at the same period of time during the day. For instance, a task or activity is the measurement of the blood pressure. And it can be done together with the weight measurement and the morning questionnaire during the morning. Thus, they all together can form up the morning context.

The scenarios detected within this system are three. The first and third consist of a set of measurements, making use of the wearable garments and portable devices at home. Besides, the user answers two questionnaires defined by the medical team. These two contexts form the indoors scenario. The exercise context

proposes a short walk to promote a healthy lifestyle and to improve cardiovascular capacity, which composes the outdoors scenario.

Finally, the third scenario sketches the professional interaction to assess the health status of their patients. The persona which describes the professional is a very busy specialist which needs to visualize the most significant information.

D. Framework Phase

The analysis of the different scenarios is carried through an iterative refined context scenario from the study of “day in the life” of the persona during the **Framework** phase.

The daily routine is flexible and configurable for each particular patient. That is to say, the professional together with his patient can create a particular scenario or routine depending on their needs and preferences. To test the system, a default routine for the user is fixed, in order to illustrate the functionality of the system.

Morning context lasts from 8 am to 12 am. When users are prepared, they take the PDA and press “Start morning activities”. The morning routine starts with a questionnaire about the quality of the sleep. Afterwards, the user is reminded to take his medication. Then, the UI assistant lets the user into the measurement of the blood pressure and the weight. Both measurements will be sent automatically to the patient device through BT.

Thereafter, the UI assistant guides the user through the body vital signals on how to use the wearable garments. The garment measures ECG and respiration. The quality of the signal is improved when the user is relaxed. Thus, the UI device shall advise the user to stay relax and it will discard the signal when it is not good enough. With it, the morning routine is finished.

Two hours after the medication, the user can perform a proposed physical exercise to improve his/her heart and physical condition. The user needs to wear the garment for the duration of the physical activity on. Before starting the exercise, the UI device will assure that the user is in a good condition to proceed with the exercise combining all signals and information from all the inputs (e.g. weight increase and answers to questionnaires). During the exercise, which can never be longer than 6 minutes, the UI device gives feedback to the user and controls through algorithms the adequacy of the exercise, combining the heart rate together with the activity level.

The last interaction with the system occurs in the evening, before going to sleep. Users will be asked some questions about their general wellbeing and will indicate the UI device that they are going to bed. Then, “the bed” will monitor users during the night.

All gathered data, processed raw signals and notifications are sent to the Back-end, for further processing and management.

The professional accesses all data through the portal. The first information that can be seen is an outline of every user emphasizing the most important events. The professionals can also consult and edit the information related to a particular user and compare the current tendencies with those of previous weeks and months.

E. Refinement Phase

Thereafter the four previous phases, the **Refinement** one finalizes with a detailed documentation about all the requirements and specifications. The main specifications are shown in detail in the next section.

III. SYSTEM SPECIFICATIONS

The following figure pictures all the elements that form the global system. This section presents the main requirements and specifications for each of them.

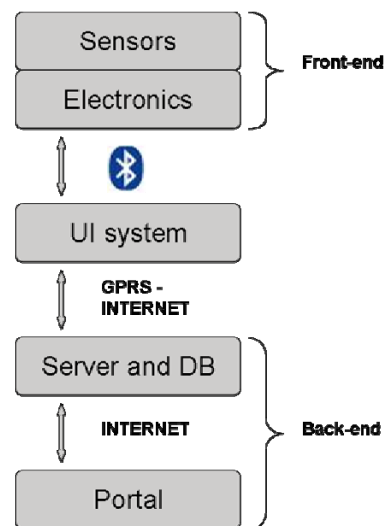


Figure 4. Global system modules

A. Monitoring sensors and devices requirements

The vital information needed by the system is: electrocardiogram (EGC), respiration activity monitoring, weight and blood pressure records.

In order to record the vital signals, wearable sensors are used. The ECG and respiration during the day are measured with a textile garment, which can be a vest or a T-shirt. The measure of the ECG is one lead with a sample rate of 256 samples/sec. The information is used to obtain heart rate (HR), heart rate variability (HRV), among others. The respiration is also measured with a

rate of 25 samples/sec. It is used for obtaining the amplitude and respiration rate.

The activity is obtained with a three dimension accelerometer placed in the on-body electronics box. This box also takes the signals from the garment, filters them and amplifies them. They also extract the heart rate from the ECG and the respiration rate and amplitude from the respiration. Such electronics also enables the Bluetooth transmission.

For the night measurement, there is a piezo foil under the sheets. The raw data obtained goes into the stationary electronics obtaining the heart rate, respiration frequency and amplitude; and activity level as outputs.

Besides the wearable and stationary sensors and electronics (both wearable and stationary) there are two other additional off-the-shelf electronics: a blood pressure cuff and a weight scale. The blood pressure cuff is an A&D UA-767PBT [6]. The weight is an A&D UC-321PBT. Both are validated in accordance with organizations such as the Association of the Advancement of Medical Instrumentation (AAMI), among others.

B. Communication requirements

The communication among sensors and electronic is wired based. From the electronics and devices, it enables a Bluetooth link with a serial port profile.

The blood pressure cuff and the weight scale follow a proprietary A&D protocol in which the device itself is the master of the connection. In the case of the MyHeart electronics the protocol is defined within the remit of the project by PITS-NV (Philips, Leuven, Belgium).

For communicating with the Back-end, the User device should have GPRS or UMTS connection to enable a client for a web service in the server side, through secure channels (e.g. HTTPS, SSL [7]).


C. User Interaction Methods and devices

The selected input method for end users is a touch screen, since it is more intuitive. The professionals will interact through a web-based application that allows ubiquity. For both users and professional, the feedback methods addressed are text, images, graphs and audio.

The user interaction device should be light weighted. However, it must have enough memory capacity to store the data gathered (at least 512 MB). Furthermore, a mobile connection is needed to communicate with the server through the Internet (i.e. GPRS or UMTS). Likewise, Bluetooth is necessary to connect to sensors.

Taking into account these requirements, a comparative analysis amongst different available devices in the market was performed. The most suitable device is a PDA with a touch screen and mobile communication capabilities: the QTek S-200 series.

TABLE II. THE QTEK S-200 FUNCTIONALITIES

	CPU: TI OMAP 850, 2x200Mhz
	OS: Microsoft Windows Mobile 5.0
	Screen: TFT 240x320, 2,8"
	RAM: 64 MBytes
	Bluetooth 2 support
	GPRS support
	Dimensions: 10,8 x 5,93 x 1,84 cm

D. Application Requirements

The user application needs to be highly user-friendly, intuitive and easy to use. Besides, it needs to implement different use cases workflows in runtime, such as: daily vital signals monitoring and feedback, questionnaires filling, reminders, checking and displaying results, and automatic data transfer among others.

Moreover, the whole system must support multilingualism: English by default, German and Spanish.

Furthermore, the application should validate all raw data received, together with the questionnaires and user behavior data. It should also be able to prevent errors and to recover from them.

Furthermore it should generate notifications, which are events with certain information three different levels. The first level is the communication notifications, which are problems with the Bluetooth or GPRS communication. The second one deals with the behavior of the user, that is, it notifies whether users comply or not with their personalized routine. It also gives out information about the schedule to perform each activity. The last level of notification is related to the health status, for instance a high hypertension, or a bad quality of rest during sleep.

D. Back-end Requirements

All data gathered should be stored in a robust system with high level security and remote access from the server.

The server supports all business logic, workflows and error handling. Furthermore it delivers all information to the portal, to be consulted by the professional team. The professional should be able to check all data, analyze trends and edit the daily routine, adjusting the thresholds, questionnaires, etc.

Once all the requirements and specifications are addressed, the HFMS is divided into 2 different systems: the User Interaction System and the Back-end System, described in the following section.

IV. RESULTS

The resultant HFMS, which globally executes and deals with all system requirements is composed of 2 systems: the User Interaction System and the Back-end, which includes the server, the data management module and the portal.

A. User Interaction System

This system addresses all the requirements for the user interaction, the communication needs for the sensors readings and the sending of all data to the back-end. Besides, this system implements the daily routine to be performed by users in order to adhere to their heart failure care protocol.

Thus, the user interaction system is divided into five main areas, described next:

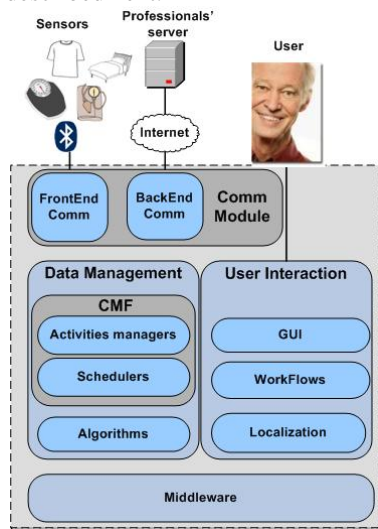


Figure 1. The Patient Station software architecture

- The communication module provides the two first areas: a) the Front-end communication module to implement the protocol stack for the communication to the Front-end devices; and b) the Back-end communication module to enable communication towards the Back-end system.
- The middleware toolset provides means to work in a comfortable and reliable programming framework.
- The data management module provides the application database, and several tools for its processing and storage.
- Finally, the user interaction module includes all formularies that implement the user interface and the application controllers to user events and interactions.

The application has been developed under Microsoft .NET framework which is widely compatible with many Pocket-PC compliant devices. The chosen programming language is C-Sharp and Compact Framework version 1 for better compatibility [8][9].

The following subsections detail each of these modules.

a) Front End Communication module

The Front End module provides a standard interface for the sensors. This module isolates the complexity of the communication protocols towards the different sensors.

The protocol towards wearable sensors allows multiple data channels, with different data types and data rates to be multiplexed on a single Bluetooth connection. The protocol is structured in layers (i.e Transmission, Framing, Multiplexing and Channel layers) in order to allow modularity and flexibility. Each layer has its own packet which encapsulates lower layer packets.

The lower layer is called the Transmission Layer. This layer just provides a bit channel over a Bluetooth serial port emulation channel. The serial port must be set on the operating system as an outbound port paired to the MyHeart sensor Bluetooth module.

The following layer is called the Framing layer and is responsible for grouping data in frames, synchronizing frames using a zero byte and making some basic error control by means of a CRC. In order to prevent zero bytes to be sent to the parts of the packet not dedicated to synchronization, COBS encoding is applied on these parts.

The Multiplexing layer multiplexes different data channels on a single physical channel. The protocol can transport up to 255 data channels plus one control channel.

The Channel layer builds up the packets containing the data and commands.

The protocol operates following a Master/Slave communication paradigm where the PDA, as the master, controls the sensor electronic, which acts as the slave.

Off the shelf sensors used in MyHeart's HFMS are developed by A&D instruments ltd. Firm. These devices use a proprietary protocol that runs over Bluetooth serial port emulation protocol. The messaging sequence is originated from the medical devices. Once the devices are paired the data are sent with a single data packet and the receiver answers with an acknowledgement packet.

b) Back-end Communication module

The Back-end module deals with the communication between the User Interaction System and the Back-end. The communication has been developed using Web Services [10] over HTTPS as the transport protocol. The module also implements security facilities like channel encryption and authentication.

The Back-end communication module deals with two kinds of data: a) data collected from the daily routine performance, such as biomedical signals and questionnaires; and b) system configuration.

All the exchanged data is structured in terms of XML language which simplifies possible protocol extensions without the need of re-implementing the objects exchanged.

This module tries to send the data when a context has been completed. When the connection fails, the module keeps on sending the data.

c) *Middleware toolset*

The middleware toolset implements four functionalities: a singleton register, an error management module, a debugging library and an inactivity watchdog.

The singleton register provides a class to store unique system elements, such as the system clock. Only one instance of these elements should exist at execution time. Therefore the singleton register is a tool for storing, allocating and de-allocating class instances.

The error management module logs the system errors during system execution.

The debugging library provides a tool for programming traces management. This helps debugging the application in case of error or fault situations and provides support to the technicians in order to resolve malfunctioning situations.

The inactivity watchdog tool is used to check user inactivity.

d) *Data management module*

The data management module covers the application data management, processing and storage.

The module comprises a Context Management Framework and an algorithm manager.

The Context Management Framework (CMF) implements and manages the Session/Context/Activity model. Application live information is stored on a data structure called *Tabloid*, inspired on the Airport Information Screens. A tabloid implements the table containing the list of actions performed and to be performed during the day by the application.

This data structure is shared all over the application so that each module can be easily aware of the application execution status. The tabloid is continuously updated by a *Scheduler*. The scheduler checks the system clock and decides the actions to be performed by the user. It manages contexts and alarms information and launches events whenever the user must be warned of some pending action.

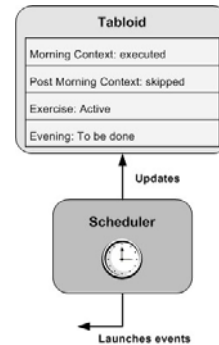


Figure 2. The Scheduler updates the Tabloid and launches events

Hence, two programming threads work within the scheduler: A scheduler thread, which updates tabloids and launches events and the main windows, and GUI thread which handles the events that activate the GUI.

Contexts execution is managed on the tabloid. Each context is defined by a starting and an ending time which sets the validity period, a set of restrictions that constrains the context execution, a set of activities to be performed, and a variable that describes the current context state.

Some possible states are described in the following table:

TABLE III. POSSIBLE STATES OF A CONTEXT

State	Description
INACTIVE	when a context has not begun yet
ACTIVE	when a context can be performed and is waiting for an event to be started
PERFORMED	when a context has been completed correctly
ABORTED	if the context execution has been aborted by the user
UNCOMPLETE	when the context has been executed, but not finished

Following this scheme, when the users' application sets off, all contexts are inactive.

Context activation follows the following rules:

1. Each context can be performed between its starting and ending time.
2. If a context has no restrictions, only the time is checked to know when to activate the context.
3. If the context has restrictions, all of them must be matched.

Once a context is activated, an alarm can be launched and can be executed with the user acknowledgement. The execution of a context means the execution of the activities of the list.

Each activity is controlled by the activity managers. For instance, a questionnaire activity manager takes care

of the extraction of questionnaires from a configuration file, fills in the questionnaire data structure and manages the answers.

The Data Management Module is also responsible for the integration, into the application, of a set of algorithms used to pre-process raw bio-medical signals.

These algorithms demand high computational and memory requirements because of the large amount of data to be processed. In order to improve the system performance, these algorithms are developed in low level programming languages such as C++ or C. Thus, the algorithms module provides a layer that supplies compatibility among algorithms libraries and the rest of the application.

e) User Interaction module

The User Interaction module manages all the interaction between the application and the user. It is responsible for generating flows of graphical forms depending on the state of the application and on external events. The module is made up of three main libraries: a set of workflows, a graphical user interface (GUI) and a localization tool.

The user interaction application flow is defined by means of workflows, in order to avoid complexity and enhance reusability.

Workflows serve as a mediator between the graphical user interface and the application data model.

The execution path is followed according to three kinds of variables:

1. user interactions: e.g. a user decides to start a context execution
2. system information: e.g. it is time to launch an alarm
3. external events: e.g. the blood cuff has sent the data

The users' application executes three kinds of workflows: a) a general application workflow which defines the main execution path, b) a context execution workflow which runs the execution of a context and c) a set of activity workflows, each one dedicated to a single activity.

The user interaction module is heavily based on event-oriented programming, each event happening, either into the activity managers or at the graphical user interface, launches an event which is managed by the workflow.

This kind of implementation requires that each formulary defines some output events, which eventually ends up closing the windows forms, and other commonly used events for skipping an activity or interrupting a context execution.

User interfaces formularies are developed using the support provided by *windows forms* provided by the Microsoft .Net Compact Framework.

The graphical user interface follows common *look & feel* for the entire MyHeart project. It is user-friendly and intuitive, with clear indications of the necessary steps to follow during the monitoring sessions. The following figure shows an example of the GUI.

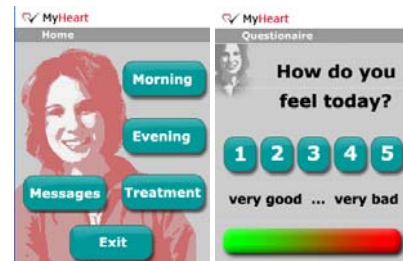


Figure 3. GUI example

The HFMS product will be delivered in English and Spanish, but will be prepared to maintain other languages such as Italian, French, German or Dutch.

In order to provide multilingualism, a localization tool has been implemented: Each form of the GUI has been deployed without any localizable information; all the texts are passed to the form class through its constructor. The localization tool is responsible for extracting the textual information depending on the users' language.

The following section describes the second main component of the system.

B. Back-end System

The Back-end comprises the Web Services for enabling communication, the data managing, the server and the professionals' portal.

The Back-end has been designed following the Model-View-Controller (MVC) philosophy [13]. The MVC model separates data (Model) from the user interface (View) concerns, so that changes to the user interface do not impact the data handling, and the data can be reorganized without changing the user interface.

The MVC design pattern solves this problem by decoupling data access and business logic from data presentation and user interaction, by introducing an intermediate component: the Controller.

The Back-end system maps to the MCV model in the following way:

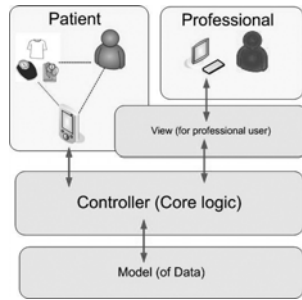


Figure 4. The MVC model into the Back-end system

The first component of the MVC architecture to be detailed is the Controller, shown in Fig. 5.

The Controller processes and responds to events, typically user actions, and may invoke changes in the model and view. It represents the core of the system that receives and processes the information sent from the patient station, and generates notifications by using different algorithms. In the HFMS' Back-end system the controller has the following architectural components:

- Communication module
- Authentication module
- Pre-Processing unit
- Device manager
- Data access library
- Notification Handler
- Analysis Unit

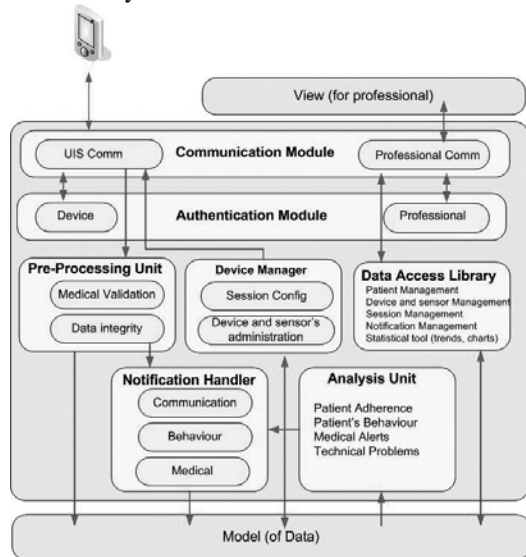


Figure 5. The Back-end Controller architecture

The communication module implements the communication towards the Patient Station and the Back-end View subsystem. Therefore, two interfaces are defined: a User Interaction System Communication that receives session data from the patient's mobile device and

uploads the session plan in the patient station, and a Professional Interaction Communication that provides a full set of functionalities to interact with the different roles of professional users (cardiologists, nurses, administrators, technicians, etc).

In order to grant confidentiality and security, the Authentication module provides the full access control to the system. Different security policies have been applied for the two interfaces of the Communication module. A further security layer has been added by using encrypted channels of communication into the Communication module.

The Back-end Controller also generates notifications, which are used to remind professionals of some important event related to a specific patient.

Notifications are generated by applying different algorithms that can be divided into three categories:

1. Communication: used to evaluate communication issues between the User Interaction System and the Back-end
2. Behavior: in order to understand how the patients use the Patient Station (adherence to the session protocol, time of use, etc). This information is also used to improve the usability of the device.
3. Medical: to assess the health status of the patient.

The Notification Handler is the entity that manages the notifications and possibly reacts to a significant event. It distinguishes notifications by their category (communication, behavior or medical), the source that generated them and the causes.

The Notification Handler is accessed from two intelligent units that process the data and generate notifications: the Pre-Processing Unit and the Analysis Unit.

The Pre-Processing Unit analyses the session data sent from the PDA. It evaluates the quality of the bio-signals and the medical coherence of the data (e.g. checks whether the weight is coherent with previous measurements). Any anomaly will generate a notification by invoking the Notification handler. Moreover the system checks if all the data has been received and notifies if any communication error should occur.

The Pre-Processing Unit is executed whenever a patient's device sends the session data. Data is then, analyzed and finally stored in the Back-end Model.

The Analysis Unit has the access to the entire data model in order to get a higher knowledge than the Pre-Processing Unit which only holds a single session's data. This module contains complex algorithms dedicated to estimate:

- The patient's adherence to the session protocol.

- The patient’s behavior relating to the use of application (time of use, velocity in executing questionnaires or other activities, etc.)
- Significant changes in patient’s health status.
- The device and sensors status in order to detect problems in the User Interaction System like low batteries, persistent communication problems, software bugs, etc.

The Analysis Unit is implemented as a daemon that checks the data periodically on the entire Back-end’s Model system.

The Data Access Library provides a simplified library for accessing the data model focused on the professionals visualization needs. The library allows retrieving composite data like overall patients data (Personal data, Treatments, Clinical history), devices data (show the device status), sessions data (session plans and acquired patients’ data), notifications, and statistical data to analyze patient’s evolution.

The Device Management implements the logic for the remote administration of the patient’s devices and sensors.

Two main functionalities are considered; the first is a configuration manager that synchronizes the session plan created by the medical staff with the patient’s device. The second is a hardware configuration used to change some options in the sensors and in the Patient Station.

This information is delivered to the User Interaction System, as a return message when it connects to the Back-end in order to send the session data.

The Model represents the domain-specific representation of the information of the application. It represents a full implementation of the data model used in the HFMS’ Back-end system.

The data model is divided into 6 main areas, presented in the following table:

TABLE IV. BACK-END MODEL AREAS

Area	Description
Professional data	It comprises the information needed for the professional access and management
Patient data	It includes all the patients’ personal data, their clinical record or their treatment history
Session Planning	It contains the planned sessions with the contexts and activities specifications
Session Data	It includes all data related to the acquired sessions (times of start and end, status of each activity, etc.). It does not contain the bio-signals
Medical Data	It comprises all the bio-signals acquired during the session performances
Algorithms data	It processes data from different algorithms
Questionnaire Model	It stores all the questionnaires

The Back-end system implementation is based on objects, while the persistent data are stored in a classical relational database. In order to fill the gap between the object orientation programming language and the relational query language, a middleware, known as Hibernate [14], has been used.

The MVC model’s View is responsible for rendering the model into a suitable form of interaction. In the Back-end the View provides the visualization for the professional users by means of a web portal and a set of logical functionalities. Following figure sketches the View architecture.

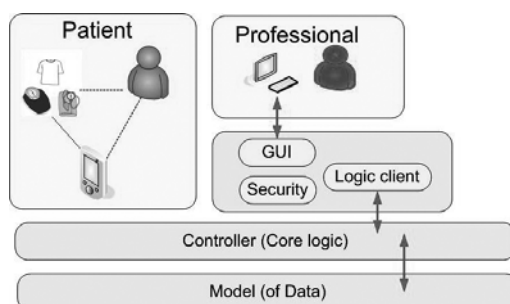


Figure 6. The Back-end View architecture

Through the View system medical staff can evaluate health’s status of the users and change their session plan.

The View comprises a GUI which implements a web interface, accessible remotely with a common browser.

Professional users have different roles within the system (administrators, cardiologists, nurses and technicians). Thus, the portal is divided into three areas for each role, as shown in following table:

TABLE V. PORTAL AREAS

Area	Description
Administration Area	Manages the users of the system, the devices, and the system logs
Medical Area	Complete GUI that manages patients, devices and helps the evaluation of patients’ health conditions. Cardiologists access all information. Nurses can use a limited version
Technical Area	Assess communication problems with the Patient Stations and sensors for the technical service

Security has been implemented considering two kinds of interactions. The first one is the one between the professionals and the View system. For this interaction, an authentication module has been developed in order to apply the roles’ policies and a secure connection over HTTPS. The second one is the interaction between the View system and the Controller. This communication channel is encrypted using HTTPS and the access to the

Controller is restricted to authorized entities using the authentication module.

The Logic client is a Web service client that invokes the available functionalities in the Back-end interface of the Controller in order to retrieve all the information the View needs to show.

The View system is implemented using Cocoon web framework of the Apache Group Foundation [15]. Cocoon allows the independency of the architecture from a given content, following the so-called “separations of concerns” (SoC) paradigm. Cocoon is engineered to provide a way to isolate three areas: the data of the application (content), the graphic design (style) and the business logic, by means of “pipelines”.

A pipeline is a series of steps for processing a particular kind of content; each component on the pipeline can provide the content, transform the data or can apply styles and producing the final output. The logic in a pipeline is implemented by setting up the right components and by designing the right connections among them. This enables a building-block approach for web solutions, hooking together components into pipelines without any required programming.

The following screenshots show the HFMS portal used by the medical professionals:



Figure 7. HFMS Portal screenshots

The preliminary results of the View system have shown that the usage of Cocoon web framework is not optimal for the purpose of this system. Hence, a new

technology is under study. Thanks to the separation of concerns, this change will not affect the other modules of the system.

IV. DISCUSSION AND CONCLUSIONS

As far as the system has been validated, these positive technical results encourage us to continue with our work and development. In particular, it has been proved that connectivity through the Internet and Web Services works as planned.

Nevertheless, more studies about security issues need to be carried out, since they are a critical part of the system. In order to create a strongly consistent system, an error handling protocol during the interchange of data will be also developed.

Hence, future work encompasses the complete technical testing, clinical validation and a complete integration of data algorithms.

As mentioned in previous section, new solutions for View system of Back-end are under study (e.g JBoss). In future releases of the system, new technologies will be integrated for further testing.

Regarding the user interaction, current work focuses on improvement of usability and minimizing interaction requirements giving the system more and more contextual awareness.

The results show promising in terms of the interaction modality implemented. However, a detailed analysis in order to enhance individuals experience and incorporate this system into their routine is still lacking. This entails an in depth study of diverse behaviour components towards e-health in order to create a tailored communication framework that boosts motivation of patients to use such systems and truly incorporate them to their daily activities. A framework to be followed would consider the analysis of different variables [16].

Finally, in the new paradigm of Ambient Intelligence, we strongly believe that in the future this kind of systems will represent an important part of the daily activity of this sort of patients, supporting a better quality of life and helping to prevent and to treat chronic diseases.

VI. ACKNOWLEDGEMENTS

This work has succeeded thanks to the close collaboration with Hospital San Carlos of Madrid, Spain. We would also like to thank the team of Medtronic, Madrid, Spain and the team from Philips Research Laboratories in Aachen, Germany, as well the one from Philips Design, Eindhoven, The Netherlands.

The Heart Failure Management System described within this paper is an integrated product of the MyHeart project, “Fighting Cardiovascular Diseases by prevention and early diagnosis” (IST-2002-507816). MyHeart

Project of the 6th Framework IST Programme, partly funded by the European Commission.

VII. REFERENCES

- [1] World Health Organization. "The Atlas of Heart Disease and Stroke". Edited by J. Mackay and G. Mensah, 2004.
- [2] Deliverable 19: "Common MyHeart Platform Requirements", MyHeart IST-2002-507816, July 2006..
- [3] M. Langham, C. Ziegeler, "Cocoon: Building XML Applications". Publisher: New Riders Publishing. 1st Edition July 19, 2002. ISBN: 0-7357-1235-2.
- [4] A Cooper, "About Face 2.0 The Essentials of Interaction Design" by Alan Cooper. Edited by Wiley Publishing, Inc. 2003. ISBN: 0-7645-26413
- [5] E. Villalba, "Preliminary Results for acceptability evaluation of a Heart Failure Monitoring System based on wearable and information technologies" presented at ICADI 2006. January, 2006 Page 301. ISBN 0-9754783-0-4.
- [6] A&D Company, Ltd. It was founded in 1977 and its headquarters are located in Tokio, Japan. More information in <http://www.telemedicine.jp/index.html>
- [7] D. Gourley, "HTTP: The Definitive Guide" O'Reilly, 2002. Chapter 14. ISBN: 1-56592-509-2.
- [8] E. Rubin, R. Yates, "Microsoft .NET Compact Framework". Edited by Sams Publishing, 2003. ISBN: 0-672-32570-5.
- [9] M. Schmidt, "Microsoft Visual C# .NET 2003". Edited by Sams Publishing, 2004. ISBN: 0-672-32580-2.
- [10] M. Gudgin, M. Hadley, N. Mendelsohn, J-J Moreau, H. F. Nielsen, "SOAP Version 1.2 Part 1: Messaging Framework", W3C Recommendation 24 June 2003, <http://www.w3.org/TR/soap12-part1/>
- [11] E. Cerami, "Web Services Essentials", Publisher: O'Reilly. Pub Date: February 2002. ISBN: 0-596-00224-6.
- [12] S. Cheshire, M. Baker, "Consistent Overhead Byte Stuffing." SIGCOMM '97, September 1997
- [13] F. Buschmann, R. Meunier, H. Rohnert, P. Sommerlad, M. Stal (1996). "Pattern-Oriented Software Architecture". John Wiley and Sons Eds.
- [14] C. Bauer, A. King, "Hibernate in Action", Manning Publications, ISBN 1932394-15-X
- [15] C. Ziegeler, "Cocoon: Building XML Applications", Paperback, 2002, ISBN:0735712352
- [16] E. del Hoyo-Barbolla, M. T. Arredondo, M. Ortega Portillo, N. Fernández, E. Villalba-Mora. A new approach to model the adoption of e-health. Proceedings 13th IEEE Mediterranean Electrotechnical Conference. Benalmádena (Spain) May 2006. ISBN: 1-4244-0088-0

Address for correspondence

Elena Villalba Mora
 Life Supporting Technologies
 ETSI Telecomunicación.D-204
 Ciudad Universitaria s/n. Madrid 28040.
 e-mail: evmora@lst.tfo.upm.es
 Tel. +34913366834 / Fax: +34913366828

E. Villalba received the Dipl-Ing (MSc) degree in Telecommunications engineering from Technical University of Madrid, Spain in 2004. In 2004, she joined Life Supporting Technologies at Technical University of Madrid, where she is currently pursuing the PhD degree. She has participated and coordinated different activities in R&D projects at the European level within the Framework Programmes. She has participated in numerous international congresses. Her research interest includes the user interaction in wearable systems as well as the software architecture design for distributed monitoring mobile systems.

M. T. Arredondo received the Dipl-Ing (MSc) degree in Electrical engineering from Technical University of Tucumán, Argentina in 1976. In 1988 she received the PhD from Technical University of Valencia, Spain. She is Full Professor of the Technical University of Madrid from 2004.

Currently she is Director of the LST (Life Supporting Technologies) group. She has a wide experience in the most challenging areas of research, such as Ambient Intelligence, m-Health, m-Social Inclusion, Info-bio-nano-cogno, Technologies, Adaptive Interfaces, Complex Virtual Reality systems, Social-Health to improve life expectancy. She's also a member of the European Union "Independent Living" forum.

M. Ottaviano received the Dipl-Ing (MSc) in Informatics engineering from Technical University of Pavia, Italy in 2005. In 2005, he joined Life Supporting Technologies at Technical University of Madrid, where he is currently pursuing the PhD degree. His research interests deal with Web Architectures, Context awareness Architectures which are included in the new paradigm known as Ambient Intelligence.

D. Salvi received the Dipl-Ing (MSc) degree in Telecommunications engineering from University of Naples Federico, Naples, Italy in 2004. In 2005, he joined Life Supporting Technologies at Technical University of Madrid, where he is currently pursuing the PhD degree. His main interest's areas are in sensor networks and decision supporting systems for the health applications.

E. Hoyo-Barbolla received the Dipl-Ing (MSc) degree in Telecommunications engineering from Technical University of Madrid, Spain in 1998. She holds an MSc in Bioengineering and Telemedicine and an Msc in information Management from 1999 and 2002 respectively. She worked for GlaxoSmithKline in their Graduate Development Programme. Currently she is a Senior Researcher and project manager at Technical University of Madrid, where she joined Life Supporting Technologies at 2003 and she is pursuing the PhD degree. Her principle area of interest is the Area of e-Health for Wellbeing, she has coordinated different activities in R&D projects at the European level within the Framework Programmes. Besides, she represents the Ministry of Education and Science in some programmes. She has participated in numerous international congresses. Her research interests lie in the area of psychological interventions and Public Health.

S. Guillen is a PhD in Telecommunications from the Technical University of Valencia. Currently he is the Director of Research at the Technical University of Valencia. He is leading the Area of Technology for Health and Wellbeing, he has directed R&D projects at the European level within the Framework Programmes, as well as technology transfer projects with national and foreign enterprises and private and public sector health care companies. The major areas of research activity focus on: E-Health, E-Inclusion, Quality of life, E-Information & Health, all of which are included in the new paradigm known as Ambient Intelligence.