UNIVERSITAT POLITÈCNICA DE VALÈNCIA

DEPARTAMENTO DE SISTEMAS INFORMATICOS Y COMPUTACIÓN

Generic Data Processing & Analysis Architecture of a Personal Health System to Manage Daily Interactive Sessions in Patients with Major Depression

MASTER THESIS IN
ARTIFICIAL INTELLIGENCE, PATTERN RECOGNITION
AND DIGITAL IMAGING (IARFID)

Presented by:
Adrián Bresó Guardado

Supervised by:
Dr. Juan Miguel García-Gómez
Dr. Juan Martínez-Miranda

Valencia – España
September 2013

Contents

Acknowledgements

First I am greatly thankful to my supervisors and friends **Dr. Juan Miguel García-Gómez** and **Dr. Juan Martínez-Miranda** for their advice in this Master Thesis and for they daily support.

Furthermore I would also like to acknowledge with much appreciation the crucial role of **Informatics Biomedical Research Group[1] (IBIME)** staff from Universitat Politècnica de València[2] (UPV), who provided me the possibility to complete this Master Thesis. All of them help me during more than three years in this and other projects with their enthusiasm, talent and sense of humour. From **Dr. Montserrat Robles** and my Master Thesis supervisors, to all my colleagues: **Miguel Esparza, Dr. Salvador Tortajada, Javier Juan, Alfonso Pérez,** and **Dr. Javier Vicente**, and specially **Carlos Sáez** and **Dr. Elíes Fuster**.

This Master Thesis has been performed within the Help4Mood FP7 European Project [ICT-248765], so I want to thank all the work done by the different partners involved in the project: Edinburgh's School of Informatics[3], I2CAT[4], Babes-Bolyai University[5], FVA[6], OBS Medical[7], Universitat Politècnica de Catalunya[8], and The School of Mathematical and Computer Sciences at Heriot-Watt University[9]. Special thanks to **Dr. Colin Matheson** and **Dr. Maria Wolters** who are the coordinators of the project.

Finally, special thanks to my **family** for their patience and support. I am especially grateful to my **wife** for all her patience and understanding, my pretty **daughter** and my near future **son**.

---

[1] http://www.ibime.upv.es/
[2] http://www.upv.es/
[3] http://www.ed.ac.uk/schools-departments/informatics/
[4] http://www.i2cat.net/es
[5] http://www.ubbcluj.ro/
[6] http://www.fvaweb.it/
[7] http://www.obsmedical.com/
[8] http://www.upc.edu/
[9] http://www.macs.hw.ac.uk/

Abstract

The World Health Organization [10] (WHO) considers **Major Unipolar Depression** as a significant cause of disability worldwide (globally, more than 350 million people of all ages suffer from depression). This common mental disorder spends many economic and clinical resources and it is sometimes responsible for patient suicides.

The work presented in this Master Thesis document describes the design and implementation of a generic **Data Processing & Analysis** module which has been included within the Personal Health System developed in the **Help4Mood**[11] Research European Project [FP7 ICT-248765]. The aim of Help4Mood is to develop a computational distributed system to support the treatment of people with Major Depression in the community. It is focused on reducing depressive symptoms, improving functioning, and preventing the recurrence of symptoms in the future.

The developed Data Processing & Analysis module is the mechanism responsible of:

i)   The analysis of the objective and subjective inputs received from the user;

ii)   The inference of clinical concepts and the production of a set of activities to be suggested by the system during the treatment of depression;

iii)   The planning of the most appropriate interactive sessions based on the inferred activities;

iv)   The generation of adequate emotional responses represented in the Help4Mood's Virtual Agent to better engages the patient with the use of the system and to facilitate a better adherence to the treatment process; and

v)   The summarization of the relevant clinical information about the progress of the patient every week.

The results of this work suggests that the generic Data Processing & Analysis module is useful for managing flexible and personalised sessions in the treatment of the depression, and it is able to be adapted to other clinical domains. It provides a systematic framework for data collection, processing and analysis which improves the continuous monitoring and treatment of the patients. Additionally, our module improves the communication between patients and clinicians by facilitating the joint reflexion about the evolution and improvements in wellbeing at the different stages of the treatment.

---

[10] World Health Organization (WHO / OMS): http://www.who.int/mental_health/mhgap/evidence/depression/en/

[11] Help4Mood Project website: http://help4mood.info/site/default.aspx

Acronyms & Abbreviations

CA	Candidate Activity
CBT	Cognitive behavioural therapy
CDS	Clinical Decision Support System
CDSS	Clinical Decision Support System
CE	Cognitive Emotional
CESD4	Center for Epidemiological Studies Depression Scale (4 items)
DA	Data Analysis
DASPS	Data Analysis and Session Planning System
DM	Dialog Manager
DMC	Daily Mood Check
DSS	Decision Support System
ECA	Embodied Conversational Agent
EEG	Electroencephalography
EG	Event Generator
EHR	Electronic Health Record
FB	Fact Base
HIS	Health Information System
HL7- CDA	Health Level 7 Clinical Document Architecture
ICE	Internet Communication Engine
ICT	Information and Communication Technologies
IE	Inference Engine
IVA	Intelligent Virtual Agents
KB	Knowledge Base
KE	Knowledge Extraction
KI	Knowledge Inference
NTQ	Negative Thoughts Questionnaire
PAM	Personal Ambient Monitoring
PA	Planned Activity
PHQ9	Patient Health Questionnaire (9 items)
PHS	Personal Health System
PMS	Personal Monitoring System
PoC	Point-of-Care
RBS	Rule Base System
RMT	Remote Patient Monitoring and Treatment
SP	Session Planner
VA	Virtual Agent
WHO	World Health Organization
WM	Working Memory
XML	Extensible Markup Language

XSD   Extensible Markup Language Schema Documentation

Chapter 1

Introduction

1.1. Motivation

Personal Health Systems (PHS) refers to the use of Information and Communication Technologies (ICT) for the improvement of healthcare. It concerns the individualization of prevention, treatment and wellbeing procedures available through the healthcare system. PHS put emphasis in the empowerment of patients at the centre of the health delivery process (citizen-centred health care) through the development and deployment of remote monitoring and healthcare management applications. It aims in bringing continuity of care at all levels of healthcare delivery, spanning from location, to ambience, to time.

PHS covers a wide range of systems including wearable, implantable or portable systems, as well as Point-of-Care (PoC) diagnostic devices. The European Commission provided the architecture of the PHSs into three main blocks as shown in Figure 1.
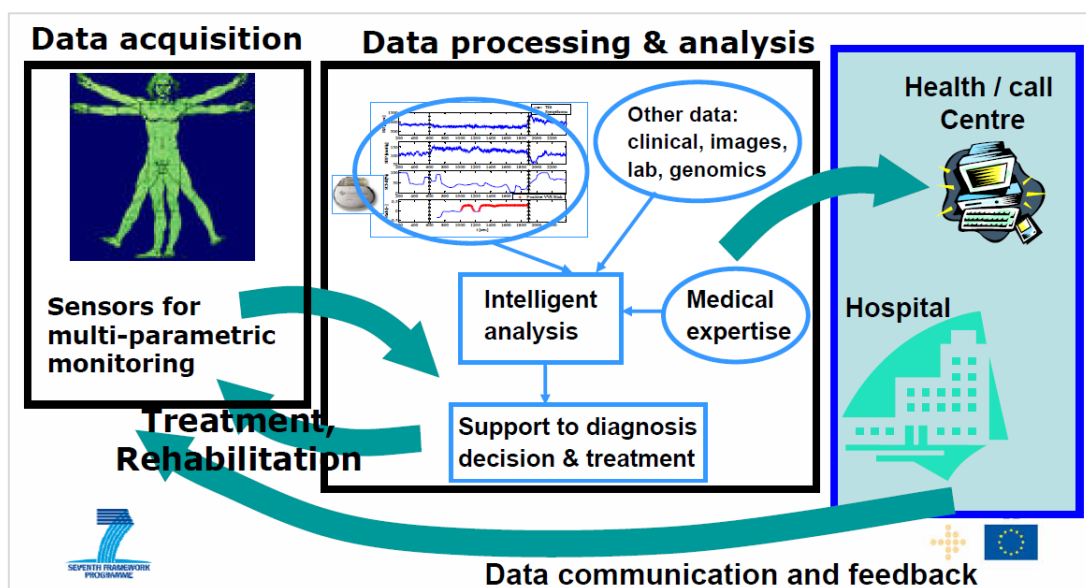

Figure 1: Personal Health System architecture provided by the European Commission.

One of the clinical domains in which PHS has been applied during the last years is in the treatment of Mental Health disorders. People with mental health disorders require continuous, quality controlled and personalised health services to empower them (especially when the suffered disorder becomes chronic). PHS offer to clinicians and patients with significant new capabilities to remotely monitor patient's conditions and facilitates the communication and immediate feedback between all the participants in the treatment process.

The World Health Organization[12] (WHO) asserts that **Depression** is a common mental disorder, characterized by sadness, loss of interest or pleasure, feelings of guilt or low self-worth, disturbed sleep or appetite, feelings of tiredness, and poor concentration. WHO's forecast shows that in 2012, 350 million people were affected by depression. In extreme cases,

---

[12] World Health Organization (WHO / OMS): http://www.who.int/mental_health/mhgap/evidence/depression/en/

depression can lead to suicide. Globally, it is estimated that suicides causes 1 million deaths every year.

Effective treatments of depression include psychosocial therapies and medication. The active involvement of depressed people and those who are close to them in addressing depression is a key issue. The first step is to recognise the suffering or depression and reach out for support. But because of the stigma that is often still attached to depression, many people fail to acknowledge that they are ill and do not seek advice for treatment [1]. The earlier the treatment begins, the more effective it is. The aim of any treatment should be to achieve remission, not merely symptoms reduction. The presence of residual symptomatology after an episode of depression increases the risk of relapse, a long-term chronic course, higher risks of suicide, poor social functioning, and poor outcome [2].

Depression interferes with the ability to function at work, school or home. It is the leading cause of disability worldwide, and it is a major contributor to the global burden of disease. This burden is divided into indirect costs (costly for the nation in reduced or lost productivity) and in direct costs (medical resources used for care, treatment, and rehabilitation). WHO suggests that the total annual cost of depression in Europe was estimated at €118 billion in 2004 (which corresponds to a cost of €253 per inhabitant per year) and it will increase to 15% in 2020. Sobocki et al. [3] asserted that depression is the most costly brain disorder in Europe.

Major efforts need to be invested in continuous patient management and monitoring over time due to the prevalence, morbidity and the economic burden of Major Depression. Hence, collaborative strategies that include the empowerment of patients are required. In order to achieve a sustainable service, active and continuous monitoring could be supported by autonomous intelligent systems with abilities to analyse the situation and react accordingly.

The purpose of this work is the development of one of the main modules of a PHS focused on the treatment of Major Depression aimed to assist in the provision of continuous and personalized Remote Patient Monitoring and Treatment (RMT). This PHS ensures a continuous monitoring of the patient's well-being based on worldwide-accepted clinical procedures applied to the specific mood conditions of the patients at each stage of the treatment. This PHS might encourage patients by means of daily personalised and interactive sessions to cooperate, provide, monitor, and understand their disease as well as improve patient-clinicians communication.

In this Master Thesis document, we describe the design, the implementation, and the evaluation of a Generic Data Processing & Analysis Architecture of a PHS to manage daily interactive sessions during treatment stages. This generic architecture can be adapted to different domains but we describe its use to the treatment of major depression in the context of the Help4Mood project.

1.2. The Help4Mood Project

The work presented in this document has been developed in the context of the Help4Mood European research project [FP7 ICT-248765] with the title: A Computational Distributed System to Support the Treatment of Patients with Major Depression.

The Help4Mood research project is aimed primarily with the improvement of the monitoring and treatment of patients with Major Depression and delayed recovery in the community [4]. An interactive system and physical sensors provide to clinicians with detailed information on a range of mental health factors assessed by the interactions with a virtual agent alongside measurements of physical activity and sleep patterns (see Figure 2).
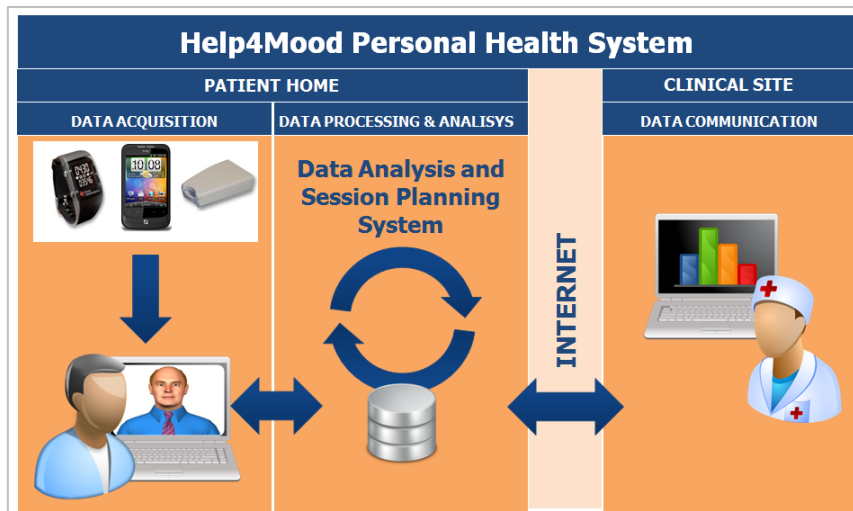
Figure 2: The general architecture of the Help4Mood system (patient and clinical sites).

The main impulse behind the Help4Mood project is the fact that depression is one of the most common causes of short and long term disability in Europe, accounting for substantial costs both directly to health services and indirectly through lost productivity and the burden of caring. Most patients recover with treatment, but for many of them, the recovery process is either slow or incomplete. Researches have shown that psychological therapies can be delivered effectively without face to face contact and that Computerized Cognitive Behavioural Therapy (CCBT) can be suitable for self-guided treatment in the individual's own home [5] [6]. However, its value is limited by the difficulty of keeping patients engaged, and some clinicians have professional concerns that important changes in mood may be missed.

Help4Mood implements a PHS aimed to offer assistance to the patient in their own home on an ongoing and systematic basis through the provision of personalised daily sessions. Each daily session is provided through a Virtual Agent (VA) and the content of the session is adapted according to the patient's data collected which include activity, demographical and historical data, as well as responses to standardised questionnaires. Figure 3 shows a screenshot of the Help4Mood interface, in which the VA is administering the PHQ9 questionnaire [7] to the patient.
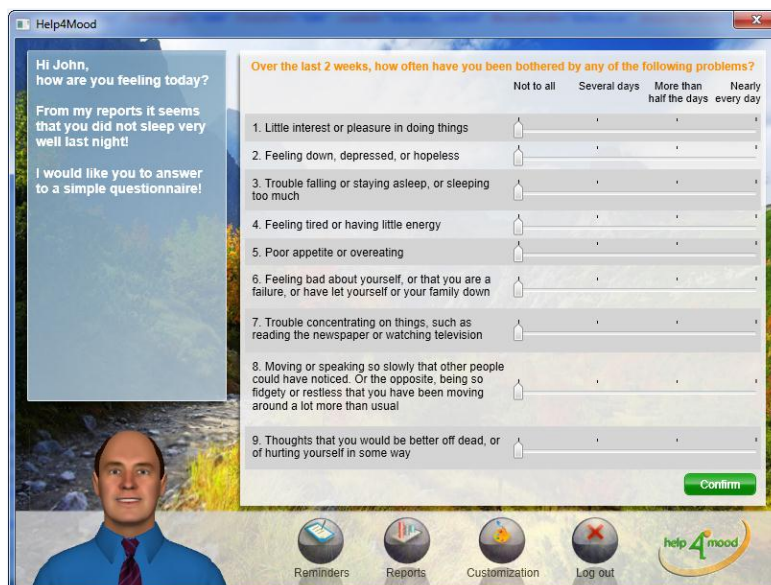

Figure 3: Screenshot of Help4Mood graphical user interface.

The Help4Mood system is composed by the following three main components distributed into the three PHS modules (see Figure 5):

- **Data Acquisition Module**

  – **The Personal Monitoring System (PMS)** that collects data about the person with Major Depression who is being supported. PMS sensors are designed to be as non-obtrusive, non-stigmatising, and suitable for shared spaces as possible (such as wrist actigraphy or smart Phones). Objective monitoring will focus on two main aspects, activity and sleep patterns. This system is described in detail in [8].

  – **The Virtual Agent (VA)** that constitutes the main patient-side user interface. Patients interact with the system through a VA which takes the form of a realistic talking head (Figure 4). The VA-patient interaction facilitates the remote and continuous questioning of specific aspects related with the mood and cognition of the patient as well as the suggestion of activities that provide information about aspects of neuropsychomotor function such as reaction time. The VA has been developed to offer a real time interaction displayed through specific dialogues complemented by a set of specific head movements (e.g. nodding, moving head up and down), facial expressions (e.g. eye brow movements) and gaze movements (see details in [9]). The rendering of the head movements and facial expressions are used to communicate an adequate emotional response to the patient's inputs (see details in [10] and in Chapter 5). The emotional responses are used to complement the interaction dialogues generated by the **Dialog Manager (DM)** component.

Figure 4: Images of the different characters implemented

- **Data Processing and Analysis Module**

  – **Data Analysis and Session Planning System (DASPS)**, which is the module developed and presented in this Master Thesis document. The DASPS integrates the functionality of a Clinical Decision Support System (CDSS) into the PHS and it manages the data collection, interpretation and assessment of the patient's wellbeing. Based on the data analysis results, the DASPS manages the daily interactive sessions between the VA and the patient which promotes a better adherence to the treatment. Moreover, the DASPS performs an assessment of the situations occurred during the interactive session and generates an adequate emotion –modelled in the VA– for the next cycle of interaction with the aim to create a better relationship between the patient and the VA. Finally, the DASPS also supports to caregivers with the interpretation of the patient's self-reported and monitored data through the generation of a weekly summary report facilitating the interpretation of patient's evolution (see Chapters 3 and 4, [11]).

- **Data Communication**

  – All the collected data and the inferred information in the DASPS are provided enveloped in HL7-CDA[13] messages to the clinical site. These data can also include the progress

---

[13] Health Level Seven International: http://www.hl7.org/index.cfm

summary report in a PDF format. The HL7-CDA format facilitates the integration of the information generated in Help4Mood into the particular Health Information System (HIS) of the clinical institution [12].
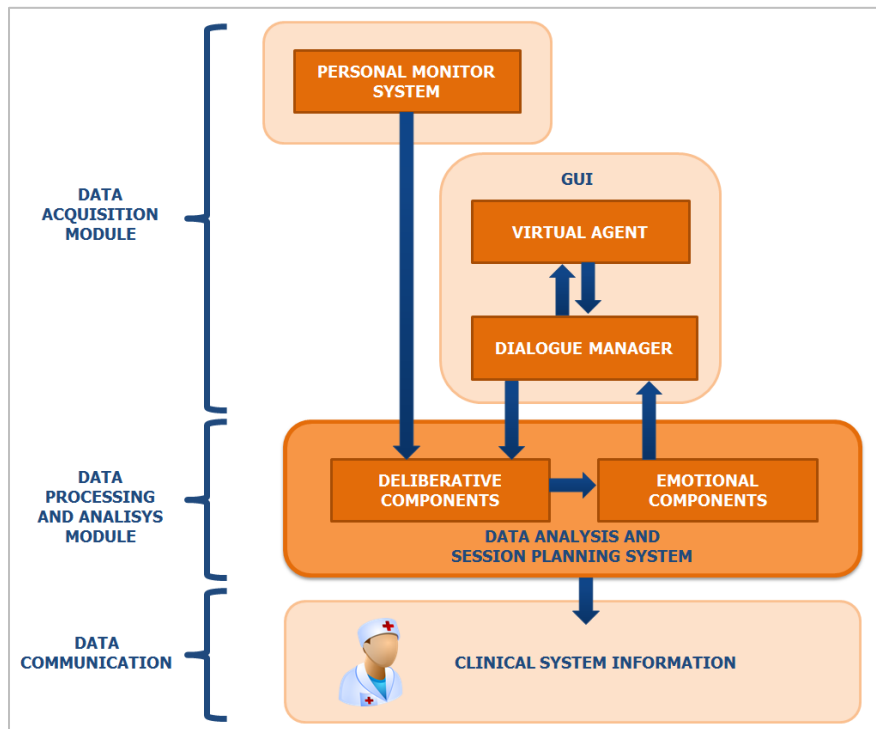

Figure 5: Architecture of the Hel4Mood Personal Health System

Due to the high modularity of the Help4Mood system, it has been necessary the definition of some data structures for the communication between components. The exchange of information includes relevant data for the Help4Mood units, such as sessions or measures derived from the monitoring devices. These data structures have been implemented using the XML format. The clinical meaning of these elements has been described, when possible, using standardised medical terminologies such as SNOMED CT [14]. When no SNOMED-CT concept exists, an internal Help4Mood codification has been created [13]. All the XML-based messages exchanged between each Help4Mood module are communicated using the Internet Communication Engine [15] (ICE) toolkit.

Eight partners have been involved during three years in the development of Help4Mood Project. Each partner has contributed with different tasks as presented in Table 1:

	Partner	**Rol**
1	THE UNIVERSITY of EDINBURGH	Edinburgh's School of Informatics was involved on: Coordination of the Project, clinical expertise, and dialogue systems
2	UNIVERSITAT POLITÈCNICA DE VALÈNCIA	**Universitat Politècnica de València was involved on: Data Processing and Analysis Module of the PHS**

---

[14] The International Health Terminology Standards Development Organization: http://www.ihtsdo.org/snomed-ct/

[15] The Internet Communications Engine: http://www.zeroc.com/ice.html

3		I2CAT was involved on: Telecommunications and dissemination activities
4		Babes-Bolyai University was involved on: Clinical expertise.
5		FVA was involved on: Graphical development of the VA and GUI of the system.
6		OBS Medical was involved on: Exploitation of the project's outcomes.
7		Universitat Politècnica de Catalunya was involved on: Personal Monitoring System.
8		HWU The School of Mathematical and Computer Sciences was involved on: Dialogue Manager.

Table 1: The Help4Mood Consortium

1.3. Objectives

In the context of the Help4Mood project the work described in this document is focused in the development of two main parts that form the DASPS of the PHS:

1. The design and implementation of a flexible and modular architecture responsible of the (1) processing and analysis of the objective and subjective data gathered from the patient; (2) the inference of clinical information that allow the continuous assessment of the patient's wellbeing; (3) the planning of personalised and adaptive daily sessions; and (4) the generation of an *easy to digest* weekly summary report to inform the clinicians about the evolution of the patient.

2. The development of a Cognitive-Emotional module which is responsible to produce adequate emotional responses in the VA during the interaction with the patient. These emotional responses simulate a therapeutic empathic behaviour in the VA when interacting with the patient.

1.4. Contributions

As a result of the Data Analysis and Session Planning System (DASPS) presented in this Master Thesis document, we highlight the following major contributions:

i. The implementation of a script which collects and extracts useful information from the patient sensors in order to detect the changes in the patient's daily activity patterns;

ii. The implementation of an Inference System containing two inter-related Rule Base Systems (RBSs). The first engine infers clinical concepts. The second infers a set of recommended activities to be suggested by the system during the treatment of depression from the clinical concepts previously inferred (and other variables);

iii. The implementation of a planning algorithm based on patient's historical data and stamina to produce the content of the personalized daily sessions for the treatment of Major Depression;

iv. The adaptation and evaluation of a computational architecture of emotions to produce adequate emotional responses in a VA during the interaction with people with Major Depression; and

v. The implementation of a summarized document based on AORTIS clinical summarization model which informs about the relevant patient progress.

Currently, some of these contributions have directly derived into the publication (or submission) of several research papers as follows:

Related Area	Publication
Affective Computing	**Modelling Therapeutic Empathy in a Virtual Agent to Support the Remote Treatment of Major Depression** Juan Martínez-Miranda, **Adrián Bresó**, Juan Miguel García-Gómez. In Proceedings of the 4th International Conference on Agents and Artificial Intelligence (ICAART). Vol 2, 264-269. Feb 2012. Vilamoura, Portugal. 2012
Affective Computing	**The Construction of a Cognitive-Emotional Module for the Help4Mood's Virtual Agent** Juan Martínez-Miranda, **Adrián Bresó**, Juan Miguel García-Gómez. Proceedings of the 1st Workshop on ICT applied to Mental Health. Spain (1) 2012:34-39. 2012
Pattern Recognition	**Monitoring changes in daily actigraphy patterns of free-living patients** Elíes Fuster-García, Javier Juan-Albarracín, **Adrián Bresó**, Juan Miguel García-Gómez. Accepted at International Work-Conference on Bioinformatics and Biomedical Engineering, IWBBIO-2013, Granada, Spain. 2013
CDSS	**A Clinical Decision Support System to Manage Daily Interactive Sessions in Patients with Major Depression** **Adrián Bresó**, Juan Martínez-Miranda, Elíes Fuster-García, Carlos Sáez, Montserrat Robles, Juan M García-Gómez. Jornada de Investigación ITACA 2013, Valencia, Spain. 2013
CDSS	**A Flexible Computational Architecture for a Personal Health System to Support the Treatment of Depression.** **Adrián Bresó**, Juan Martínez-Miranda, Juan Miguel García-Gómez. 2013 In preparation
iHealth	**Help4Mood - Supporting Joint Sense Making in the Treatment of Major Depressive Disorder** M. Wolters, C. Burton, C. Matheson, **Adrián Bresó**, A. Szentagotai, J. Martinez-Miranda, E. Fuster, J. Rosell, C. Pagliari y B. McKinstry. Workshop on Interactive Systems in Healthcare (WISH), Washington DC , 2013. Under review

Table 2: Publications related with the Master Thesis

1.5. Contents

In the following **Chapter 2** we present a review of the state of the art in the research topics related with the work developed in this Master Thesis: the field of Clinical Decision Support Systems (CDSS) as the basis of the DASPS architecture, complemented with a brief review of other related Personal Health Systems. Moreover, a review of some works in the field of Virtual Emotional Agents is also presented as the roots of the cognitive-emotional component described in **Chapter 5**. The design and implementation of the different modules of the DASPS are described in detail in the **Chapters 3, 4, 5 and 6**. In particular, **Chapter 3** presents an

overview of the DASPS architecture and the different modules that are integrated within it. **Chapter 4** shows the modules that form the deliberative process of the DASPS while **Chapter 5** shows the module that produce the emotional process. The design and development of the clinical summary report as a relevant output of the DASPS is presented in the **Chapter 6**. **Chapter 7** summarises the evaluation and results obtained from the real pilots. Finally, the conclusions of this Master Thesis are described in the **Chapter 8**.

Chapter 2

State of the Art

2.1. Decision Support Systems

Clinical Decision Support Systems (CDSS or CDS) are interactive systems that assist physicians and other health professionals with making clinical decisions based on a given set of health information, such as determining a patient's diagnosis [14].

Fully computerized CDSS typically consist of three parts; Knowledge Base (KB), Inference Engine (IE), and User Interface (see Figure 6). The KB contains rules and associations which most often take the form of "*if-then*" rules. The IE combines the rules from the KB with the patient data. The resulting analyses are displayed through the User Interface, which also takes input from the user that can result in new queries and exploration of alternative scenarios.
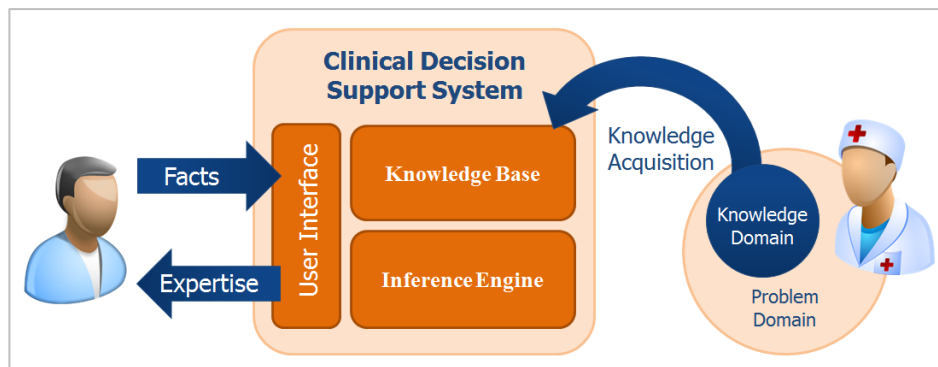

Figure 6: Typical CDSS architecture

There is growing recognition that CDSS holds great potential to improve health care quality, possibly even increase efficiency and reduce health care costs, and reduce patient inconvenience [15]. CDSS can provide support to clinicians at various stages in the care process, from preventive care through diagnosis and treatment to monitoring and follow up. It can be focused in a lot of clinical domains, such as diabetes [16], Heart Failure [17], or mental health disorders [18] [19] [20] [21] [22] [23] [24] [25].

In the field of mental health, a comprehensive search of PubMed and Google Scholar identified three early stage DSS projects to improve diagnosis, one implementing a standardized interview [25], and two based on machine learning, one from EEG [24] and one from clinical input [22] [23]. For the treatment, management or follow up of the Major Depression disorder there are currently two commonly used different approaches: Medication or Psychotherapy as the Cognitive-Behaviour Therapy (CBT).

The main algorithms for adjusting patients' medications during the treatment of Major Depression that have been reported in the literature are the TMAP algorithm [20] and its successor, the STAR*D algorithm [26]. The Texas Medication Algorithm Project (TMAP) guidelines are based on serial mood monitoring at 2-4 weekly intervals and was drawn up by consensus in 1999 [27]. The TMAP algorithm has been successfully implemented in a CDSS [20]. According to Trivedi and Daly's review of CDSS for the TMAP and STAR*D guidelines [28], the main role for a CDSS in this context is to ensure health care practitioners can easily access the algorithm and obtain correct information about suggested next steps at the point of care.

Fortney et al. [21] developed a web-based decision support system (NetDSS[16]) that supports depression care manager activities based on evidence. These activities are addressing on education and activation, self-management, symptom monitoring, counseling adherence, medication adherence, etc. Additionally, this system includes standardized questionnaires (such as PHQ9 or Blessed Orientation-Memory-Concentration) in order to evaluates de depression level and the comorbidities (such as panic disorder, anxiety disorder, alcohol use disorder).

The developed DSS of Fortney [21] is similar than the DASPS module described in this Master Thesis but we added an analysis of activity data collected with actigrapy devices (not only with the subjective data provided by the patient through the user interface), a flexible and dynamic daily session planning, and the generation of adequate emotional responses to the patient's inputs during the daily sessions.

CCBT systems are fully focused to support CBT. These self-help systems have the potential to increase the capacity of mental health services, and to overcome some of the barriers to accessing mental health services, including stigma, traveling time for rural patients, treatment delays, and the low availability of skilled clinicians [29]. Examples of available CCBT packages for the treatment of depression include Beating the Blues [17] [30], Cope [31], MoodGym[18] [32] and ODIN[19] [33]. There are several studies in which the efficiency of the CCBTs is evaluated [34] [35] [36] [37]. In general, the CCBTs improve the effectiveness in the treatment and reduce the symptoms of depression.

In the context of this Master Thesis, we will extend the definition of CDSS to cover intelligent, adaptive, interactive monitoring: existing health information is used to determine and to decide what other pieces of self-reported health information need to be collected in order to arrive at a comprehensive view of a patient's mental and emotional state. This functionality will be the cornerstone of the patient-side DASP module as one of the key features of Help4Mood. To our knowledge, this is the first CDSS that assess the patient data, infers the adequate clinical concepts, infer a set of activities to suggest configuring a personalised daily session and complemented with an emotional response based on the events produced during the interaction. Nevertheless, it is important to notice that our DASPS module is focused CBT-like therapy interaction and at the moment it does not implement support for pharmacological treatment of Major Depression.

2.2. Personal Health Systems

Health care systems are changing their approach from reactive (based on managing a person's disease) to preventive (based on managing a person's health) in order to provide high-quality and safe care, to reduce economic burden, to improve the competitiveness and interoperability leading to technology advances, and to educate and to empower patients. Personal Health Systems (PHS) allow assisting in providing continuous and personalized health services to individuals regardless of location to improve lifestyle management and prevention, early diagnosis, treatment, and disease management [38]. Additionally, PHS reduces costs in the medium-long term.

We can see the PHS as an evolution of the CDSS. Personal Health Systems have evolved towards ubiquity, connectivity and information sharing. Therefore, PHS are CDSS with added layers that allow a continuous connectivity (both patients and clinicians with), communication and better data acquisition.

In the literature, there is a good number of PHS developed and integrated into different clinical domains, such as diabetes [39]. One particular domain in which different PHSs are recently focused is in the treatment of mental health disorders. In particular, we can find in the literature several examples such as P4Well (Pervasive Personal and PsychoPhysiological

---

[16] Depression CSS: https://www.netdss.net/

[17] Beating the Blues CCBT: http://www.beatingtheblues.co.uk/

[18] MoodGym CCBT: https://moodgym.anu.edu.au/welcome

[19] ODIN CCBT: http://www.kpchr.org/feelbetter/

management of WELLness) [40], which applies CCBT methods focused in stress. Krefting et al. [41] shows a PHS that acquires bio-signals from the patient during the sleep period in order to diagnosis and treat sleep disorders.

Moreover, quite a few EC research projects [42] have been focused in the treatment of mental health disorders. In particular OPTIMI[20], MONARCA[21], PSYCHE[22] and ICT4Depression[23] propose similar objectives to Help4Mood for the remote monitoring and treatment of Major Depression and bipolar disorder. The OPTIMI project is concerned with innovative monitoring technologies, in particular electroencephalography (EEG). Other modalities include voice analysis, physical activity monitoring, and computerised diaries. ICT4Depression emphasizes medication adherence. Monitoring is mainly based on mobile phones and wireless bio-signal sensors; urinalysis to determine the amount of medication taken is also considered. Unlike Help4Mood, which focuses on unipolar Major Depression, the MONARCA project is mainly concerned with monitoring the activity of people with bipolar disorder. Devices used include a mobile phone and a wrist worn activity monitor, galvanic skin response using a custom sensor integrated into a sock and also periodic measurements of EEG. They also track GPS location, nutrition, household activity, amount and quality of sleep and voice analysis from mobile phone conversations.

Like MONARCA, PSYCHE has a particular focus on bipolar disorder. However, their multi-parametric monitoring system is based on innovative textile platforms and portable sensing devices. In addition to voice analysis, heart rate variability, respiratory rate, activity, sleep pattern, and peripheral measures in cardiovascular and respiratory functioning, PSYCHE also tracks biochemical markers, electro dermal response, as well as the secretion of stress-related hormones. Finally, the UK project Personalised Ambient Monitoring[24] (PAM) develops monitoring approaches for people with bipolar disorders. They use sensors situated in patient's home to collect information on light and sound levels, movement information, and aspects of TV usage. Information about activities and mood is collected via a mobile phone interface.

There are some differences between the Data Processing and Analysis Module of the PHS presented in this Master Thesis regarding the others PHS referred above. Our Data Processing and Analysis Module provides a high and standardized module integration, a clinical concepts inference, an adequate emotional response, and a flexible and dynamic planning of daily interactive sessions

2.3. **Emotional Virtual Agents**

Virtual Agents (VAs) or Intelligent Virtual Agents (IVA) are a combination of virtual reality and agent technology that allow users to interact with the machine through complex (such as natural speech, gestures, facial expressions, and gaze) or simple (such as questionnaire) procedures, engaging face-to-face relationships. VAs are autonomous and intelligent entities in virtual environments with appearances as human, animal or fictional characters.

For several decades, the growing interest in the development of VAs has encouraged the improvement in the realism of graphics and the agent behaviour. One of the most important improvements is the modelling of emotions as part of the VA's behaviour (Emotional Virtual Agents). Picard et al. [43] said that the main benefit of providing intelligent agents with emotions is that agents are able to simulate the understanding of human emotions producing less frustrating systems to interact with the user, facilitating human-machine interface and increasing the believability of the system. Well-designed VAs have demonstrated better mechanisms to motivate users [44] and lead to higher user engagement than traditional static visual user interfaces [45].

---

[20] OPTIMI European Project: http://www.optimiproject.eu/
[21] MONARCA European Project: https://monarca.dk/
[22] PSYCHE European Project: http://www.psyche-project.org/
[23] ICT4Depression European Project: http://www.ict4depression.eu/
[24] PAM European Project: http://www.pam-research.org/

The development of *virtual counsellors* is one of the applications of virtual agents that has concentrated good efforts in recent years. In the treatment of mental health disorders is necessary to create an intimacy relationship between the patient and the clinician to achieve a better sharing of personal information. Some works as the proposed by Kang and colleagues et al. [46] showed a study in which associated intimate self-disclosure and nonverbal VA behaviour during counselling interactions. They demonstrated that appropriate VA nonverbal behaviour got highly intimate self-disclosure and the final session results would be better.

In the Help4Mood project, the VA is an Embodied Conversational Agent (ECA) with the aim to establish an adequate relationship with the user to promote a better treatment adherence. Currently, there are many promising examples of ECA-based interfaces in mental health applications. Bickmore et al. [47] developed an agent who encouraged people with schizophrenia to adhere to their medication regime, as well as an agent who provided discharge information to people with depression who were about to leave the hospital where they had been admitted for treatment. Pontier et al. [48] showed that it is feasible to administer a standardised questionnaire for assessing depression, in this case the Beck Depression Inventory, using an ECA. Rizzo [49] and colleagues lead the SimCoach project[25], in which a web-based virtual human healthcare helps military personnel and family members in multiple areas including depression, stress, brain injury, substance abuse, suicide, rehabilitation, reintegration and other relevant specialties offering expert advice and healthcare information. The main goal of SimCoach is engaging the user for gathering user healthcare information, understanding their situations, monitoring their healthcare, giving advices and assistance, and recommending initiate a traditional treatment when needed.

A common characteristic of these VAs used in mental health applications is the modelling of adequate emotional responses that enhances the interaction with the user. The production of a consistent emotional behaviour in VAs is achieved through the modelling of emotional processes based on psychological theories of emotions. The most relevant emotions theories are summarized in Scherer et al. [50] [51]: Adaptation, Dimensional, Appraisal, Motivational, Circuit, Basic emotion, Lexical, and Social constructivist. The selection of a specific emotion theory as the basis of a computational model depends on the aspects of emotion that the computational model tries to represent. Some computational models of emotion combine theories to represent different components and phases of the emotion phenomenon.

The theory of emotion that is most often used as a basis for computational models of emotions is the cognitive appraisal theory [52] [53]. Appraisal theories of emotions assume that the events in a person's environment are constantly evaluated by the individual. This *cognitive* evaluation (or appraisal) process leads to an emotional response (according to the events relevance for the person) which in turn generates a specific behaviour to cope with the appraised events. One of the reasons for the success of this theory in computational modelling is that it provides an explanation for the connection between cognition and emotion. This, in turn, helps in the construction of artificial systems that simulate complex human-like behaviours. Moreover, appraisal theories of emotion appear to be the most comprehensive way to represent the complexity of the emotion process, covering the whole path from low-level appraisals of the eliciting event to high-level influence over behaviour [54]. The proposed cognitive-emotional module developed for Help4Mood and presented in Chapter 5 is based on this cognitive appraisal theory of emotions.

Several computational architectures based on the appraisal theory of emotions (see Figure 7) have been developed since the early 90's (Ar [55], FAtiMA [56], Alma [57], MA [58], among some others), most of them focused on the construction of believable characters within virtual environments. The development of virtual characters based on appraisal theories has allowed the creation of real-time interactive characters that exhibit emotions in order to make them more compelling, more realistic, more able to understand human motivational state, or more able to induce desirable social effects in the users.

---

[25] http://ict.usc.edu/prototypes/simcoach/
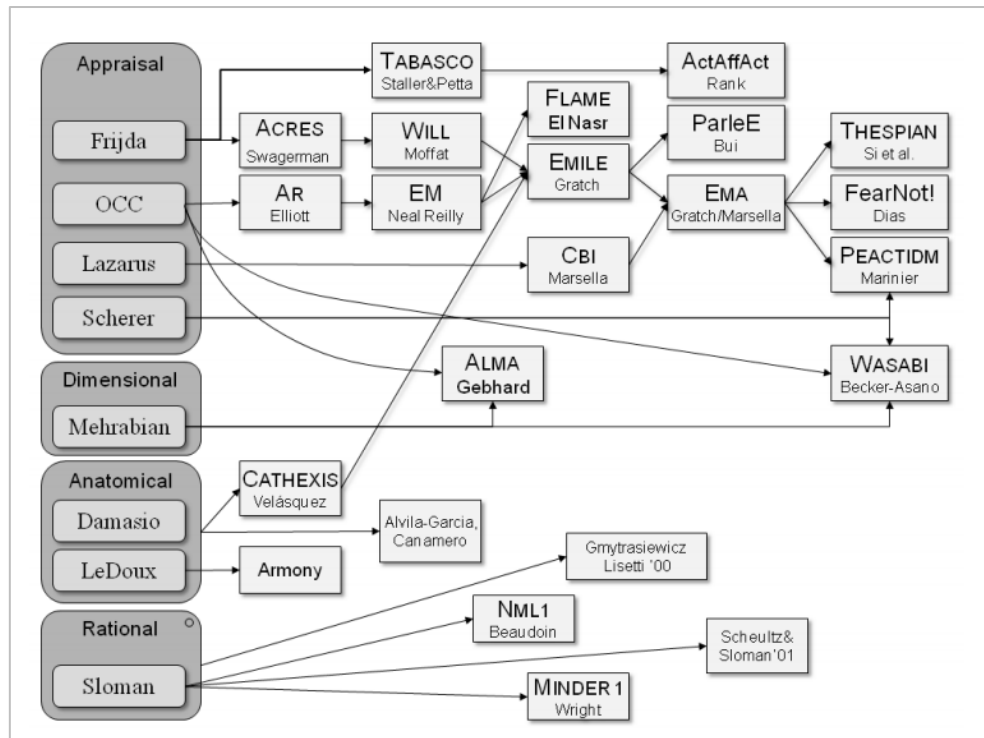
Figure 7: A history of computational models of emotions (from **[59]**).

The emotional component (i.e. the mechanism to produce the adequate emotional responses in the VA during the interaction with the user) of the DASPS module has its roots in the appraisal theory of emotions. In particular, one of the existent computational architectures of (FAtiMA – FearNot!) emotion has been adapted and integrated with the deliberative component to produce the whole cognitive and emotional behaviour in the Help4Mood's VA as it is explained in the following chapters.

Chapter 3

The Data Analysis and Session Planning System

The Data Analysis and Session Planning System (DASPS) is one of the three main components of the Help4Mood project. It implements the Data Processing & Analysis layer of the PHS. It has been conceived as the module responsible to analyse relevant patient's data, plan the daily interactive sessions and recommend the activities as part of the treatment of Major Depression. Moreover, the interaction style displayed through the dialogues and through the non-verbal communication in the VA is another functionality implemented in the DASPS module which would facilitate an active involvement and communication of both the patient and the clinician during specific stages of the treatment.

3.1. DASPS Functionalities

The main functionalities of the DASPS are:

   i. The data analysis for the identification of activity and sleep patterns applied to the data coming from the PMS;
   ii. The inference of clinical concepts and the inference of tasks recommended the patient during the daily sessions as part of the treatment;
   iii. The control of the interactions between the patient and the VA by planning the different activities which the VA offers during the daily sessions. These activities will gather information of specific mood-related data that complements the data collected through the PMS and will provide relevant feedback and recommend basic CBT-based exercises to the patient;
   iv. The generation of a set of different emotional reactions according with the events produced during the session. These emotional reactions will be expressed through particular phrases in the dialogue (managed by the DM) and displayed via facial expressions and head movements (managed by the graphical component of the VA); and
   v. The summarization of a clinical report of the patient's health and well-being to the clinician acting as a decision aid during the treatment based on the collected information during the daily sessions. This report will highlight the main trends in the patient's mood, behaviour, cognitive and psychomotor function, either graphically (e.g. activity and mood graphs over time) or as a short textual summary.

3.2. General Architecture

Based on the Help4Mood project requirements, the main development tasks of the DASPS were concentrated in the design of a flexible architecture capable of inferring and producing the appropriate events and actions which occur during a session to lead the behaviour of the VA. The development of a scalable architecture (Figure 8) has been developed thinking in an easy customization and extension with the addition of new future functionalities and its adaptation to other PHS used for the treatment of a different disease.

Six components have been developed into the DASPS, all of them integrated as a Java packages. The modules (which are showed in Figure 8) are:

   • **Data Analysis (DA)**, which collects and extracts useful information from the PMS in order to detect the changes in the patient's daily activity patterns.

- **Knowledge Extraction (KE)**, which receives and processes the input data (from DA and VA) transforming it into SNOMED-CT (or internal) concepts, and assessing the patient responses (good, normal, bad).
- **Knowledge Inference (KI)**, which infers a set of appropriate clinical activities (called Candidate Activities) recommended by the clinicians based on historical and acquired data.
- **Session Planner (SP)**, which adapts the contents and duration of the daily sessions based on patient's stamina (Long, Medium and Short sessions).
- **Events Generator (EG)**, which parse the selected activity by the SP (called Planned Activity) into specific Events to be appraised.
- **Cognitive Emotional (CE)**, which appraises the Event to generate an adequate emotion to be displayed by the VA during the interaction and the next task offered to the patient.

We can classify the components of the DASPS into two main groups: **Deliberative and Emotional processes**. In the Figure 8, the red components are the components in charge of the deliberative processes whereas the blue component is the component in charge of the emotional process. All of these components are described in detail in the next sections.
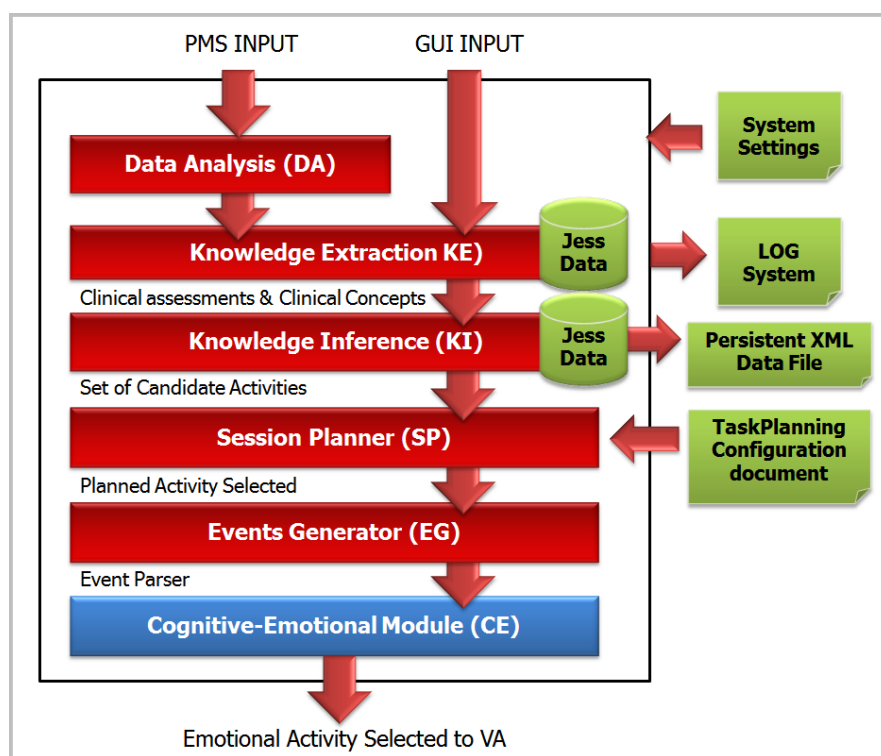

Figure 8: The Help4Mood DASPS Architecture.

Additionally, we developed the **Manager Report Generator Module** (described in Chapter 6). This module is neither of type deliberative or emotional but it implements a complementary functionality. This module is responsible of gather the relevant information from different sources (such as JesData.xml file and Data Analysis module) in order to provide one of the most important DASPS output: a Clinical Patient Progress Report. The Manager Report Generator module follows the AORTIS [60] model in order to carry out the clinical data summarization process.

All the exchange of information between the DASPS modules is performed using XML-based messages which are validated using XSD schemas. This communication facilitates the easy integration of new modules –if required– without perform internal changes in the existent modules but only adapting the communication API implemented in the messages.

Chapter 4

Deliberative Components

4.1. The Inference System

The Inference System (see Figure 9) analyses the data coming from the PMS, the patient's responses to the questionnaires provided by the VA, and the historical data (persistent file) in order to infer the appropriate set of Candidate Activities (CAs) which would be planned in the next iteration cycle of the daily session.

Three modules were developed in the Inference System: (1) **Data Analysis** which was implemented using Matlab; (2) **Knowledge Extraction** and (3) **Knowledge Inference** which were implemented using Java 7.

We defined XSD templates for the xml messages received from the PMS, from the GUI (Annex B), from the persistent file, and for the internal messages which are exchanged between the modules.
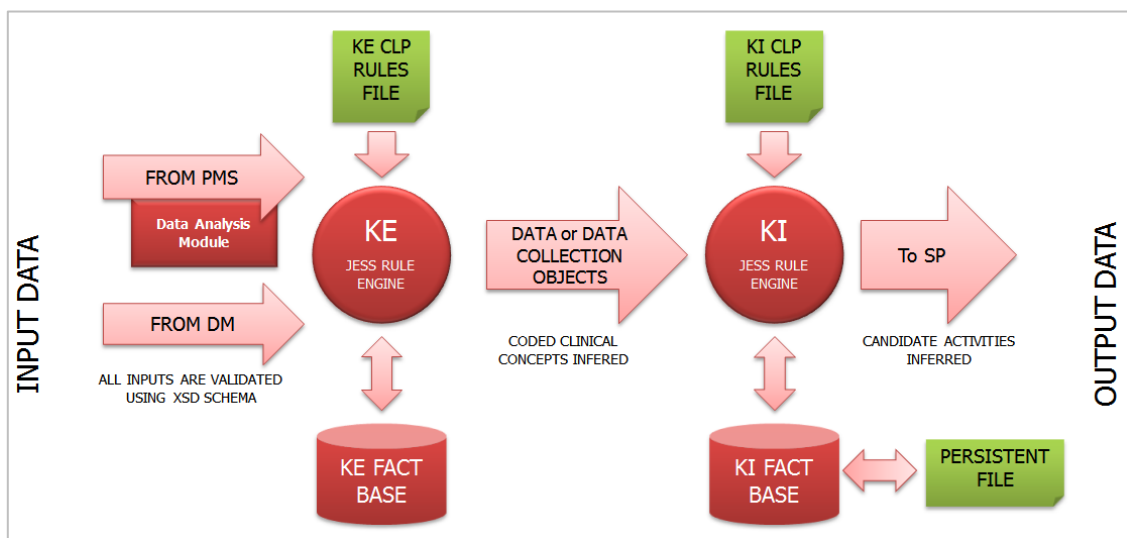
Figure 9: Inference System Architecture.

4.1.1. The Data Analysis Module (DA)

The first module in the inference system (and in the pipeline of the DASPS) is the **Data Analysis Module** (DA), which is responsible of the analysis of the patient's activity data coming from the PMS. This information is received in a XML-based structure (see Figure 10). It may contain information from different sensor devices. Each sensor is codified using a unique identifier.

```
<?xml version="1.0" encoding="UTF-8"?>
<ns1:activity xmlns:ns1="activity.data.ke.dss.h4m.bmg.ibime.upv.es"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="urn:activity.data.ke.dss.h4m.bmg.ibime.upv.es
./config/config-dss/docs/activity.xsd">
<device>
    <id>12901597530001300015_001</id>
    <ts>2012-05-18 14:23:15</ts>
    <data>
        <act><ts>2012-03-31 07:17:10</ts><val>9</val></act>
        <act><ts>2012-03-31 07:18:10</ts><val>4</val></act>
        <act><ts>2012-03-31 07:19:10</ts><val>0</val></act>
        <act><ts>2012-03-31 07:19:13</ts><val>3</val></act>
        <act><ts>2012-03-31 07:20:13</ts><val>0</val></act>
        <act><ts>2012-03-31 07:21:13</ts><val>0</val></act>
    </data>
</device>
<device>
    <id>11673452920004900025_001</id>
    <ts>2012-05-18 14:23:15</ts>
    <data>
        <act><ts>2012-03-31 07:17:10</ts><val>1</val></act>
        <act><ts>2012-03-31 07:18:10</ts><val>0</val></act>
        <act><ts>2012-03-31 07:19:10</ts><val>0</val></act>
        <act><ts>2012-03-31 07:19:42</ts><val>1</val></act>
    </data>
</device>
</ns1:activity>
```

Figure 10: XML message from the PMS to DA

Using the actigraphy device (wrist watch, phone, or a key ring) of the PMS, an initial analysis has been performed to study how to monitor and detect changes in daily physical activity patterns to try to contribute to a better assessment of the progress of the patient's condition. This analysis allows the identification of daily activity models for specific groups of days, such as working days, weekends, or days in which the patient is under specific stages of the treatment in order to detect possible crisis or relevant events in the future. Full implementation of this algorithm was described in [61].

Moreover, from the analysis of actigraphy data, a wide variety of parameters are calculated by the DA module for each day of study such as:

- **Mean daily activity**. This parameter represents the average of the actigraphy signal values along all waking periods.

- **Standard deviation of daily activity**. This parameter represents the standard deviation of the actigraphy signal values over all waking periods.

- **Maximum sustained activity**. This parameter characterizes the maximum sustained activity over a period of 30 minutes during the whole day. This value is defined as the maximum value of the daily actigraphy signal filtered using a moving average filter with a span of 30 minutes.

- **Total hours of sleep**. This parameter represents the total sleep time detected in a period of 24 hours.

- **Sleep fragmentation**. Sleep fragmentation is a parameter defined as the number of periods of uninterrupted sleep during 24 hours.

- **Mean activity during sleep**. This parameter corresponds to the mean value of the activity signal in the detected sleep periods.

- **Total time of missing data**. This parameter represents the total missed data time detected in a 24 hour period.

In addition to this information, the DA module also generates three plots which are using in the report generation task (see Chapter 6):

- **Monitored data** (Figure 11), which is a 2D representation that allows the visual study and comparison of the daily patterns (sleep quality vs. daytime activity) in a scatter plot. This plot includes the information of (1) sleep quality (based on questionnaires or actigraphy quantification), (2) day time activity, (3) lost data of each day (circle radius), and (4) connections between days. It is an intuitive, easily interpretable, and informative visualization of daily actigraphy patterns used by the clinicians to assess the patient's condition evolution.
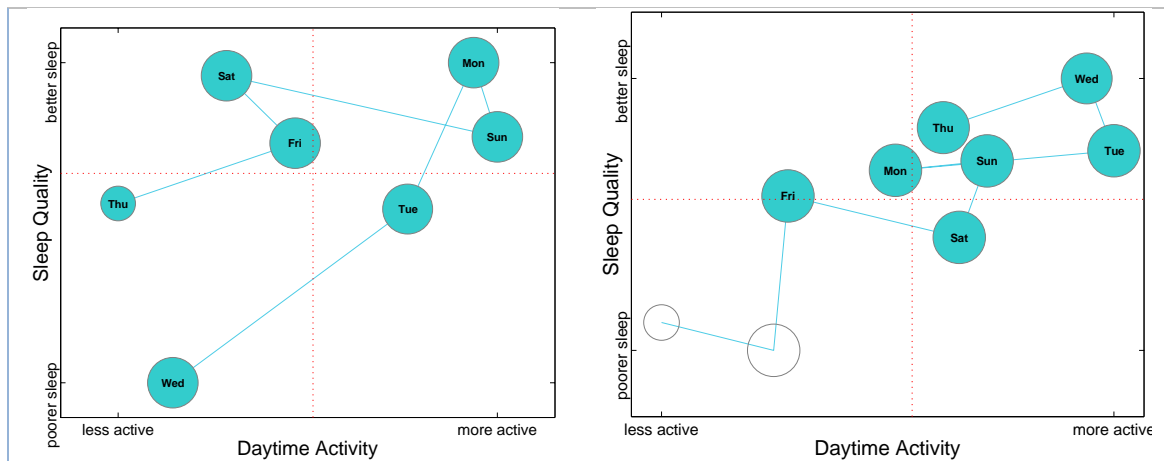

Figure 11: Examples of the plot representing monitored physical activity.

- **Mood Records** (Figure 12), which shows the evolution of the scores of the Daily Mood Check (DMC) questionnaire based on [62]. The colour bars of the graph represent the obtained values in the questions "*How is your mood today?*" (DMC1) for the "*Overall*" label, "*I have felt happy*" (DMC2) for the "*Happy*" label, "*I feel that people dislike me*" (DMC3) for the "*Disliked*" label, "*I have felt like crying*" (DMC4) for the "*Crying*" label, and "*I have talked less than usual*" (DMC5) for the "*Talkative*" label. It showed the values in a scale from 1 to 100, where 1 represents "little" and 100 "much".
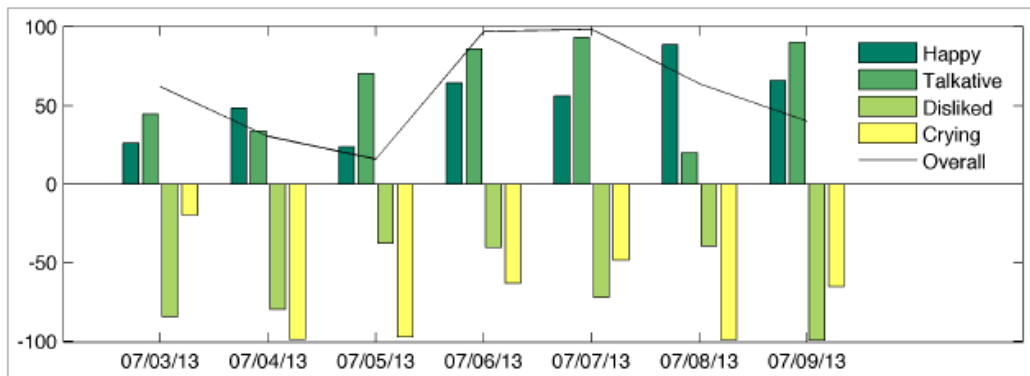

Figure 12: Daily Mood Check plot

- **Activity Record** (Figure 13), which includes the daily or weekly time expended in awake periods, long resting periods, and missed data. This plot shows easily patient adherence to the use of the sensors, and the distribution between sleep and activity period.
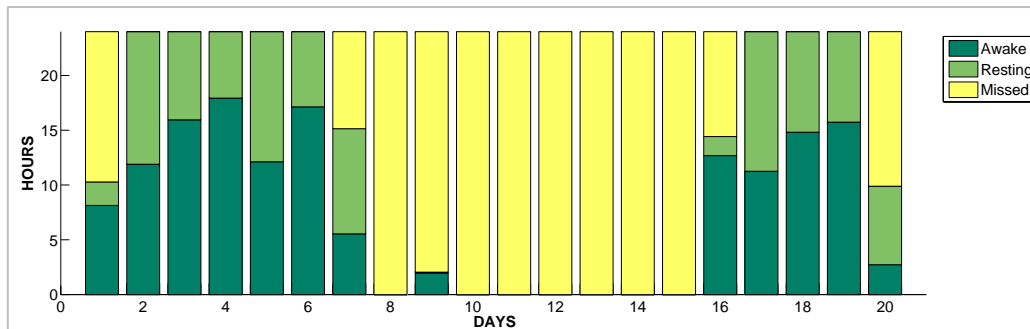
Figure 13: Activity Report plot

4.1.2.   Knowledge Extraction Module (KE)

The Knowledge Extraction (KE) module implements a Rule Base System (RBS). The RBS consists of 3 parts: a Knowledge Base (KB), an Inference Engine (IE), and a Facts Base (FB) (or Working Memory). In our work, the KB represented the expert knowledge gained from the problem domain using "*if-then*" rules. Clinicians of the consortium provided us all of this clinical knowledge. Regarding the IE, we chose the JESS Rule Engine[26] because it is fast, stable, and tightly integrated with the Sun's Java platform. The FB of the KE is the information received in XML format from the GUI (XSD template of this message is showed in Annex B) and the PMS modules.

The content of the KB is temporary and it is restarted at each iteration cycle. Some of the information in the KB is sent to the KI module (i.e. the inferred concepts) and the other information is removed (the internal control data) every iteration.

The KE module is responsible for the processing of the DASPS inputs to create the set of knowledge concepts to be used by the KI module, and for assessing the patient responses. These generated concepts represent, based in the inter-operability terminology of the project, the current patient's measures and self-reports obtained from the sensor data and from the VA. The KE uses an underlying inference-engine and data-processing methods to generate these concepts. We show in Table 3 some examples of concepts used in the Inference System.

Concept	Terminology	Code
Mild Depression	Snomed	310495003
Moderate Depression	Snomed	310496002
Severe Depression	Snomed	310497006
Restless Sleep	Snomed	12262002
PHQ9_Completed	Internal	80000003
Activity_Sleep_Time	Internal	30000402
Logout_Button_Press	Internal	80000307

Table 3: Concepts used by the Inference System

More than 100 rules have been implemented in the CLP rule file of the KE as we show in the next examples (Figure 14 and Figure 15):

```
;;--------------------------------------------------------
;; RULE KE_ReceivedFromDM_Selection_Select_Session_Type
;;--------------------------------------------------------
;;Se evalúa la respuesta del paciente ante la selección de duración de sesión
(defrule KE_ReceivedFromDM_Selection_Select_Session_Type

    ?l <- (Data{name == "Select_Session_Type"} {value == "OK"} (idSession ?l_IdSession))
    ?ll<-(Data{name == "Select_Session_Type_Selection_0_fromCode"} (idSession ?l_IdSession))
(value ?value_name) (value2 ?value_code))
    (test (eq ?l_IdSession (Utils.getIdSession)))
    (not(executeRule (rule "KE_ReceivedFromDM_Selection_Select_Session_Type")))

  =>

    (if (eq ?value_code "80000196") then ;;Long Session
        (Utils.setKE_User_Response_Assessment "GOOD")
    )
    (if (eq ?value_code "80000197") then ;;Medium Session
        (Utils.setKE_User_Response_Assessment "NORMAL")
    )
    (if (eq ?value_code "80000198") then ;;Short Session
        (Utils.setKE_User_Response_Assessment "NORMAL")
    )
    (assert (executeRule (rule "KE_ReceivedFromDM_Selection_Select_Session_Type")))
)
```

Figure 14: Example I of a rule defined in the KE.clp file

```
;;--------------------------------------------------------------
;; RULE KE_feedbackFromDailyMoodCheck_1
;;--------------------------------------------------------------
;;Se evalúa la respuesta a la cuestión 1 recibida desde el DM
(defrule KE_feedbackFromDailyMoodCheck_1

    ?l <- (Data{name == "DailyMoodCheck_1"}(value ?value) (idSession ?l_IdSession))
    (test (eq ?l_IdSession (Utils.getIdSession)))
    (not(executeRule (rule "KE_feedbackFromDailyMoodCheck_1")))

    =>

    ;;Actualizamos el valor de KE_User_Response_Assessment
    (if ( <= ?value ?*THRESHOLD_VALUE_1*) then
        (modify ?l (value2 "BAD"))
        (Utils.setKE_User_Response_Assessment "BAD")
    else
        (if (and( > ?value ?*THRESHOLD_VALUE_1*) ( < ?value ?*THRESHOLD_VALUE_2*)) then
            (modify ?l (value2 "NORMAL"))
            (Utils.setKE_User_Response_Assessment "NORMAL")
        else
            (modify ?l (value2 "GOOD"))
            (Utils.setKE_User_Response_Assessment "GOOD")
    ))

    (assert (variable (name "DailyMoodCheck_1") (value "Complete")))
    (assert (totalScoreQuestionnaire (questionnaireName "DailyMoodCheck") (totalValue ?value)))
    (assert (executeRule (rule "KE_feedbackFromDailyMoodCheck_1")))
)
```

Figure 15: Example II of a rule defined in the KE.clp file

4.1.3.   The Knowledge Inference Module (KI)

The structure of the KI is similar to the KE but the FB of the KI is filled with the inferences of the KE and the stored information of the persistent file (called JessData.xml).

The output of the KE is sent to the KI in order to continue the DASPS process pipeline. This information is used by the KI in order to infer the possible set of Candidate Activities (CAs) to execute during the session. More than forty tasks are defined in the system (see Annex A). The template of the CA is able to store: the type of activity, the type of rule, the current session, patient information, evidence, pathology, description, generation time, bibliography, preconditions, post conditions, and parameters. The typeOfActivity attribute is very important because it would change the content of the daily session (as we describe in Section 4.2). CAs are classified into:

Type of rule	Description
Execution (code: 3)	Normal activity.
Interruption (code: 4)	This must be executed immediately, clearing the current plan. Normally, this kind of activities finishes the system execution.
Bridge (code: 5)	Similar than interruptions but it allows the normal execution of the rest of the plan after its execution.
Control (code: 6)	Special rule which is used for managing some aspects of the SP

Table 4: Type of Candidate Activities

More than 140 rules have been implemented in the CLP rules file of the KI as we show in the next examples (Figure 16, Figure 17 and Figure 18):

```
;;------------------------------------------------------------
;; RULE launchProposeReportGeneration
;;------------------------------------------------------------
(defrule launchProposeReportGeneration
"If last lastReport was generated for DAYSWITHOUTSENDREPORT days"

    ?l <- (Data{name == "login"}(name ?nameLogin)(OBJECT ?l)(date ?dateLogin))
    ?ll <- (Data{name == "lastReport"}(date ?datelastReport))
    (not(executeRule (rule "launchProposeReportGeneration")))

    =>

    (if (> (Utils.diffDays ?dateLogin ?datelastReport) ?*DAYSWITHOUTSENDREPORT*) then
        (bind ?cact2(new CandidateActivity))
        (bind ?cactDescr2(new Description))
        (?cactDescr2 setName "Generate_Report")
        (?cactDescr2 setCode 10000102)
        (?cactDescr2 setSnomed "NO")
        (?cact2 setDescription ?cactDescr2)
        (?cact2 setTypeOfActivity 3)
        (add ?cact2)
    )
    (assert (executeRule (rule "launchProposeReportGeneration")))
)
```

Figure 16: Example I of a rule defined in the KI.clp file

```
;;------------------------------------------------------------
;; RULE proposePHQ9_checkDate
;;------------------------------------------------------------
(defrule proposePHQ9_checkDate
"Test whether last PHQ9 was done more than 15 days ago. If so, ask again for PHQ9."

    ?l <- (Data{name == "lastPHQ9"}(date ?dateLogin))
    (test(> (Utils.diffDaysFromToday ?dateLogin) 7))
    (not (executeRule (rule "proposePHQ9_checkDate")))

  =>

    (bind ?cact(new CandidateActivity))
    (?cact setTypeOfActivity 3)
    (bind ?cactDescr(new Description))
    (?cactDescr setName "Introduce_PHQ-9")
    (?cactDescr setCode 10000200)
    (?cactDescr setSnomed "NO")
    (?cact setDescription ?cactDescr)
    (add ?cact)

    (assert (inference(task "PHQ9")))
    (assert (executeRule (rule "proposePHQ9_checkDate")))
)
```

Figure 17: Example II of a rule defined in the KI.clp file

```
;;------------------------------------------------------------
;; RULE UserAgreeWithIntroductionPHQ9
;;------------------------------------------------------------
(defrule UserAgreeWithIntroductionPHQ9
"If the user accepts the Introduction_PHQ9 task, this rule launch PHQ9 activity."

        ?l <-(Data{name == "Introduce_PHQ-9"} {value == "OK"}(idSession ?l_IdSession))
        (not (executeRule (rule "UserAgreeWithIntroductionPHQ9")))
        (test (eq ?l_IdSession(Utils.getIdSession)))
        ?xx <- (variable (name "current_Session_Time") (value ?current_Session_date))
        ?x <- (Data{name == "Session_Time"} (date ?current_Session_date) (value ?val)
(idSession ?l_IdSession))

   =>

        (bind ?cact(new CandidateActivity))
        (?cact setTypeOfActivity 3)
        (bind ?cactDescr(new Description))
        (?cactDescr setName "PHQ9")
        (?cactDescr setCode 10000201)
        (?cactDescr setSnomed "NO")
        (?cact setDescription ?cactDescr)
        (add ?cact)
        (retract ?l)
        (assert (inference(task "PHQ9")))
        (assert (executeRule (rule "UserAgreeWithIntroductionPHQ9")))
        (modify ?x (value (Utils.getDateDiff ?current_Session_date)))
)
```

Figure 18: Example III of a rule defined in the KI.clp file

Given the good integration between JESS and Java, both CLP files (KI.clp and KE.clp) use Java classes structures in order to manage the data which we imported to the CLP files as follows:

```
(import es.upv.ibime.bmg.h4m.dss.data.commontypes.CandidateActivity)
(import es.upv.ibime.bmg.h4m.dss.data.commontypes.Description)
(import es.upv.ibime.bmg.h4m.dss.data.commontypes.Parameters)
(import es.upv.ibime.bmg.h4m.dss.data.commontypes.ParameterConcept)
(import es.upv.ibime.bmg.h4m.dss.data.commontypes.ParameterRule)
(import es.upv.ibime.bmg.h4m.dss.jess.data.*)
(import es.upv.ibime.bmg.h4m.dss.jess.Utils);
```

Figure 19: External imports from CLP files

Additionally, we defined some Jess templates in both CLP files (KI.clp and KE.clp) to customize internal and temporary data as follows:

```
(deftemplate questionFeedback (slot questionnaire) (slot questionNumber (type INTEGER)) (slot value
(type INTEGER))(slot evaluationFeedback))
(deftemplate questionnaireFeedback (slot questionnaire) (slot value (type INTEGER)) (slot
evaluationFeedback))
(deftemplate inference (slot task))
(deftemplate executeRule (slot rule))
(deftemplate variable (slot name) (slot value))
(deftemplate totalScoreQuestionnaire (slot questionnaireName) (slot totalValue) (slot
evaluationFeedback))
```

Figure 20: Internal templates defined in the CLP files

The KB of the KI reloads the historic data from the persistent file every session and updates this data until the end of the current interactive session. So the KI collects in this persistent file all the valuable information that is restored in the next daily session. The persistent file is a XML-based file (that uses the corresponding XSD schema) in which we store all the relevant information of the patient generated during each daily session as follows:

```xml
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<JessData xsi:schemaLocation="C:|Users|ADRIAN|workspace|H4M_DSS/config/config-
dss/docs/jessData.xsd" xmlns="urn:data.jess.dss.h4m.bmg.ibime.upv.es"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
<data name="Sleep_Questionnaire_Complete" value="OK"
        date="2013-07-19T13:10:32.699+02:00" code="80000200" ontology="Internal"
        value2="" idSession="8rrf7tihadijdh0q4pd1kgna39"/>
<data name="DailyMoodCheck_1_Completed" value="40"
        date="2013-07-19T12:52:26.298+02:00" code="80000116" ontology="Internal"
        value2="NORMAL" idSession="j132cc2haoj137hjl36uhuouou"/>
<data name="Report_Visualitation_Completed" value="Not Defined"
        date="2013-07-22T18:06:16.704+02:00" code="80000164" ontology="Internal"
        value2="Response from DM" idSession="v81taa8vuusmrch1fkdj0blamc"/>
<data name="Select_Session_Type_Selection" value="Long_Session"
        date="2013-07-22T14:04:00.616+02:00" code="80000199" ontology="Internal"
        value2="80000196" idSession="1lp0a2hkfj2f8487e6mqihlucp"/>
<data name="lastReport" value="OK" date="2013-07-22T18:07:09.593+02:00"
        code="80000165" ontology="Internal" value2="Update value by the System"
        idSession="vjtahh8hls2245nka8a1030j8t"/>
<data name="Activity_Sleep_Time" value="347.0"
        date="2013-07-22T18:06:47.396+02:00" code="30000402" ontology="Internal"
        value2="20130625" idSession="vjtahh8hls2245nka8a1030j8t"/>
</dataCollection>
</JessData>
```

Figure 21: JessData.xml file

4.2. The Session Planner Module (SP)

The Session Planner Module is the mechanism responsible for planning the patient's activities during the daily sessions. The SP is dynamic and flexible because is able to adapt the daily session in each iteration. It is able to make a new plan if necessary by adding, removing and/or changing the order of the activities in real time.

The SP plans the more adequate daily session to the patient based on:

- **Clinical requirements**: The clinicians can update the configuration file (TaskPlanning.xml) in order to define the set of available activities, the periodicity of the activities and the priority of the activities to be offered to the patient. This configuration file allows the clinician (or technical) to customize the daily sessions according to the particular needs of each patient (even for each stage of the treatment) for each medical centre, and for each clinician (according to his/her preferences).
- **Patient stamina**: Stamina refers to the ability to maintain prolonged physical or mental effort. We obtain the patient stamina from a GUI question: "What length of session would you prefer? Choose one option: (1) Long session, (2) Medium length session, (3) Short session".
- **Past daily sessions**: The SP stores all the sessions in order to manage the periodicity of the executed activities.
- **User feedback on the current session**: Some user responses can modify the current content of the session such as the activities refused or the score obtained in the questionnaires.

The first step in the pipeline (see Figure 22) of the SP module is the **interruption detector**. The SP looks for a bridge or interruption (e.g. the execution of a Crisis plan triggered by a requesting of the user in the GUI or triggered by a low score detected in one of the questions of the PHQ-9 questionnaire) into the inferred set of CAs received from the Inference System. If the SP finds some interruption, the SP will cancel the current plan and it will select the interruption as the next task to execute. If it finds a bridge, the SP select the bridge as the next task to execute but it doesn't remove the current plan.
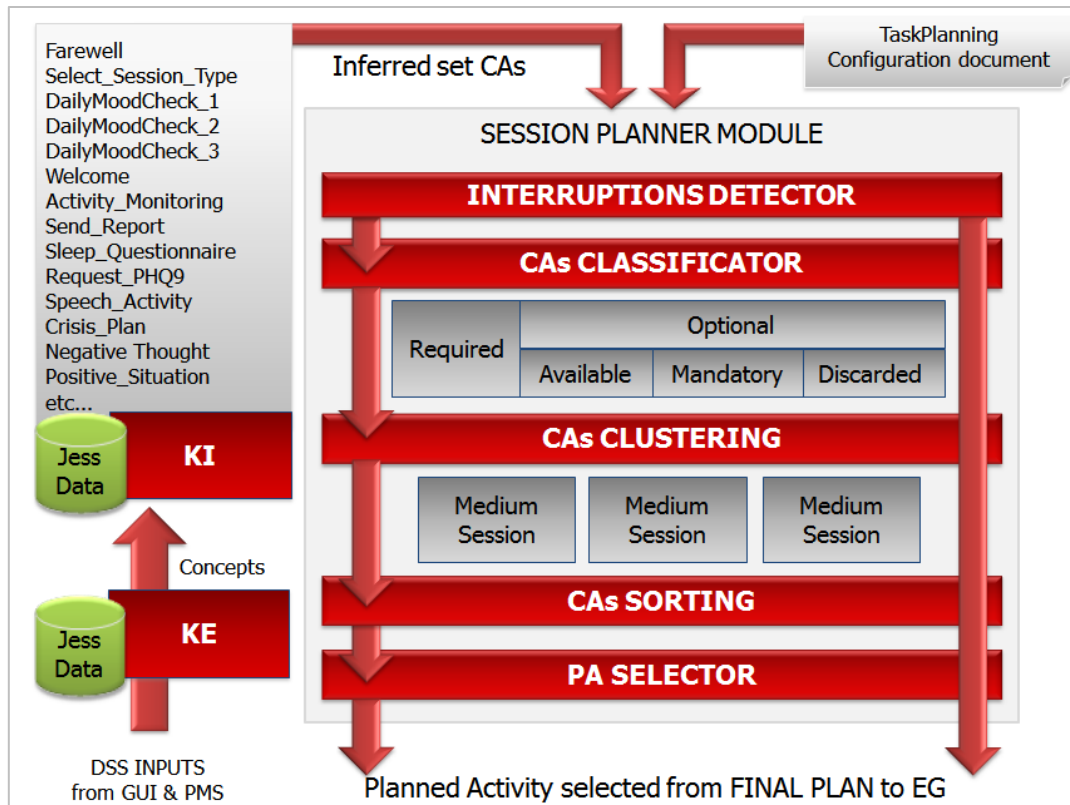
Figure 22: Pipeline of the Session Planner Module

In all the other cases, the SP will continue executing the pipeline and will **classify the Candidate Activities** into:

- **Required**: All the required activities received from the Inference System must be added as content of the actual session. Examples of this type of activity include the daily mood check questionnaire (to be executed in every session) the displaying of the summary report or the PHQ-9 questionnaire (which are executed weekly).

- **Optional**: These CAs can be included or skipped from the content of the actual session depending on past sessions and on the pre-defined values associated with this type of activity: **minimum frequency** and **maximum frequency**. These frequencies represent how often the activities need to be executed within a week. Examples of these activities include the speech analysis (which can be executed e.g. at least once per week and a maximum of three times per a week) or the monitoring of negative thoughts (which can be executed e.g. at least three times per week and a maximum of seven times per week, i.e. daily). Additionally, this kind of CAs are classified into:

  - **Available**: whose minimum number of executions (minFrec) can be still achieved in the remaining days of the week. So, <u>we CAN include it in the current plan</u>.

maxFrec of task > number of executions of the task this week	(1)

  - **Mandatory**: that necessarily needs to be included in the current session to reach the minimum number of executions (minFrec) in a week. So, <u>we MUST include it in the current plan.</u>

minFrec of task > number of executions of the task this week && (minFrec of task - number of executions of the task this week) <= rest of the days	(2)

- o **Discarded**: that reached the maximum number of executions in a week (maxFrec) and is not considered for the current session. So, <u>we MUST REMOVE it from the current plan.</u>

maxFrec of task <= number of executions of the task this week	(3)

In order to carry out this classification, the SP needs the information of the TaskPlanning.xml (see XSD template in Annex C) file. In the following example, there are defined two CAs, one defined as *Required* and other defined as *Optional*.

```xml
<item>
<description>
        <code>10000100</code>
        <snomed>NO</snomed>
        <name>Report_Ready</name>
</description>
<enable>YES</enable>
<typeOfItem>REQUIRED</typeOfItem>
<priority>100</priority>
<minFrec>0</minFrec>
<maxFrec>0</maxFrec>
<codeOfDataInDB></codeOfDataInDB>
<repeatable>NO</repeatable>
<constrains>
        <constrain>
                <constrainDescription>
                        <code>10000012</code>
                        <snomed>NO</snomed>
                        <name>Welcome_Reminder</name>
                </constrainDescription>
                <constrainType>BEFORE</constrainType>
        </constrain>
        <constrain>
                <constrainDescription>
                        <code>10000011</code>
                        <snomed>NO</snomed>
                        <name>Welcome</name>
                </constrainDescription>
                <constrainType>BEFORE</constrainType>
        </constrain>
        <constrain>
                <constrainDescription>
                        <code>10000102</code>
                        <snomed>NO</snomed>
                        <name>Generate_Report</name>
                </constrainDescription>
                <constrainType>ENABLEBY</constrainType>
        </constrain>
        <constrain>
                <constrainDescription>
                        <code>10000121</code>
                        <snomed>NO</snomed>
                        <name>Farewell</name>
                </constrainDescription>
                <constrainType>AFTER</constrainType>
        </constrain>
</constrains>
</item>

<item>
<description>
<code>10000300</code>
<snomed>NO</snomed>
```

```
<name>Speech_Activity</name>
</description>
<enable>YES</enable>
<typeOfItem>OPTIONAL</typeOfItem>
<priority>20</priority>
<minFrec>3</minFrec>
<maxFrec>5</maxFrec>
<codeOfDataInDB>80000130</codeOfDataInDB><!--Speech_Complete-->
<repeatable>NO</repeatable>
<constrains>
        <constrain>
                <constrainDescription>
                        <code>10000012</code>
                        <snomed>NO</snomed>
                        <name>Welcome_Reminder</name>
                </constrainDescription>
                <constrainType>BEFORE</constrainType>
        </constrain>
        <constrain>
                <constrainDescription>
                        <code>10000011</code>
                        <snomed>NO</snomed>
                        <name>Welcome</name>
                </constrainDescription>
                <constrainType>BEFORE</constrainType>
        </constrain>
        <constrain>
                <constrainDescription>
                        <code>10000121</code>
                        <snomed>NO</snomed>
                        <name>Farewell</name>
                </constrainDescription>
                <constrainType>AFTER</constrainType>
        </constrain>
</constrains>
</item>
```

Figure 23: Example of *Report_Ready* and *Speech_Activity* Candidate Activities definition in the TaskPlanning.xml file

After the classification, the next step in the pipeline (see Figure 22) of the SP module is the **Clustering** process. The SP clusters the classified CAs into three groups: Long Session, Medium Session, and Short Session.

The resulting classification of the CAs allows the SP to carry out the different group sessions using the following criteria:

- **Short Session**: all mandatory CA.

- **Medium Session**: all mandatory CA + (available CA)/2.

- **Long Session**: all mandatory CA + available CA.

The SP algorithm to cluster the session is as follows:

```
For each candidateActivity in setOfCandidateActivities {
    If (candidateActivity.getState(taskState.mandatory))
    {
        sessionMedium.add(candidateActivity);
        sessionLong.add(candidateActivity);
        sessionShort.add(candidateActivity);
    }
    If (candidateActivity.getState(taskState.available))
    {
        sessionLong.add(candidateActivity);
```

```
        If (alternate) {
                sessionMedium.add(candidateActivity);
                alternate=false;
        }
        else{alternate=true:}
    }
}
```

Figure 24: Algorithm to cluster CAs in different sessions

After the clustering, the SP module sorts the CAs using the information of the TaskPlanning.xml file. Besides the frequency (minFrec and maxFrec) and other attributes, clinicians can establish in this xml file some values such as the priority of the activity or the order constrains (AFTER, BEFORE, ENABLEBY or BLOCKBY) in order to carry out the CAs sorting task. The SP creates all the possible sequences which satisfy the pre-defined constrains. After this, the SP selects the best combination using the priorities as a discriminant.

Finally, the first CA of the generated plan is selected (and deleted from the plan) in order to send it to the next module of the DASPS, the Event Generator Module. The selected CA is called Planned Activity (PA).

In order to better understand the pipeline of the SP Module, three different scenarios are represented in the following Figure 25, Figure 26, and Figure 27:
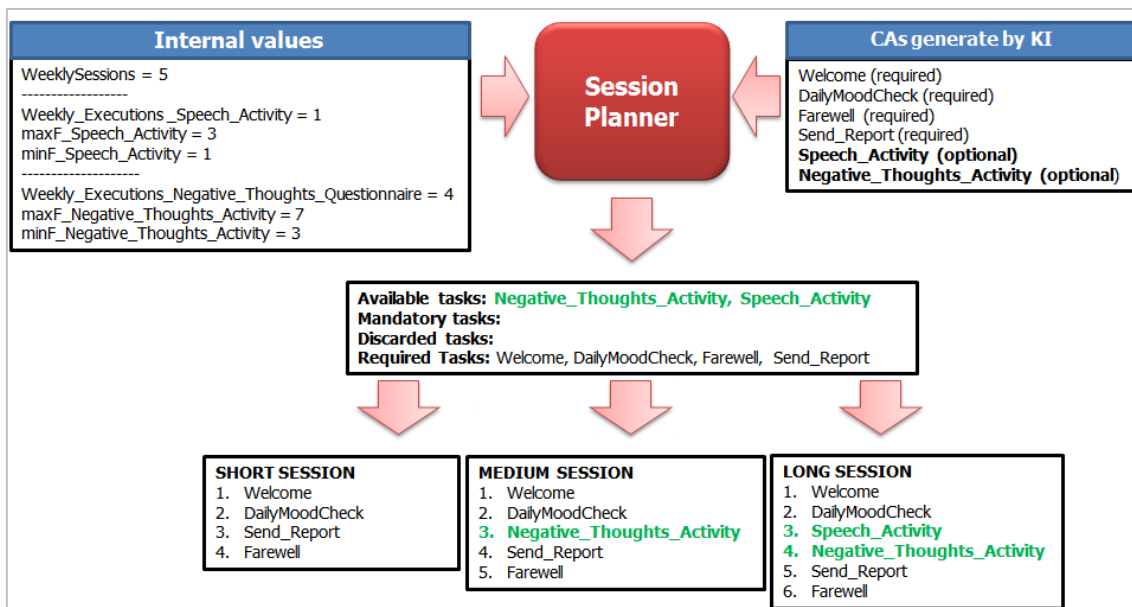
Figure 25: Example I of the SP Module execution.

In the first scenario (Figure 25), the SP receives six CAs from the KI. Four of them are required, so they are included in the actual plan. Other two CAs are Optional and we need to classify them into available, mandatory or discarded to make the different session configurations.

Regarding the Speech_Activity, the SP gathers from the historical data that this activity has been completed once during the current week. The defined minFrec of this CA is 1 and the maxFrec is 3. So the SP meets the goal and the SP can plan it more times this week because it is under the maxFrec limit. So, the SP classifies the Speech Activity as Available. Regarding the Negative_Thoughts_Activity we found the same case, the SP can plan it more times during the current week and then it is labelled also as Available.

In order to cluster the different configurations of the sessions, the SP uses only the required CAs to get the Short Session. To get the Medium Session, the SP uses the required CAs and only one available activity (the one with the major priority). Finally, to get the Long Session, the SP uses all the required and available CAs.

In the second example (Figure 26) the Speech_Activity has been completed twice during the current and the value of the maxFrec is set to 2. So the SP must not plan this CA and it is labelled as discarded.
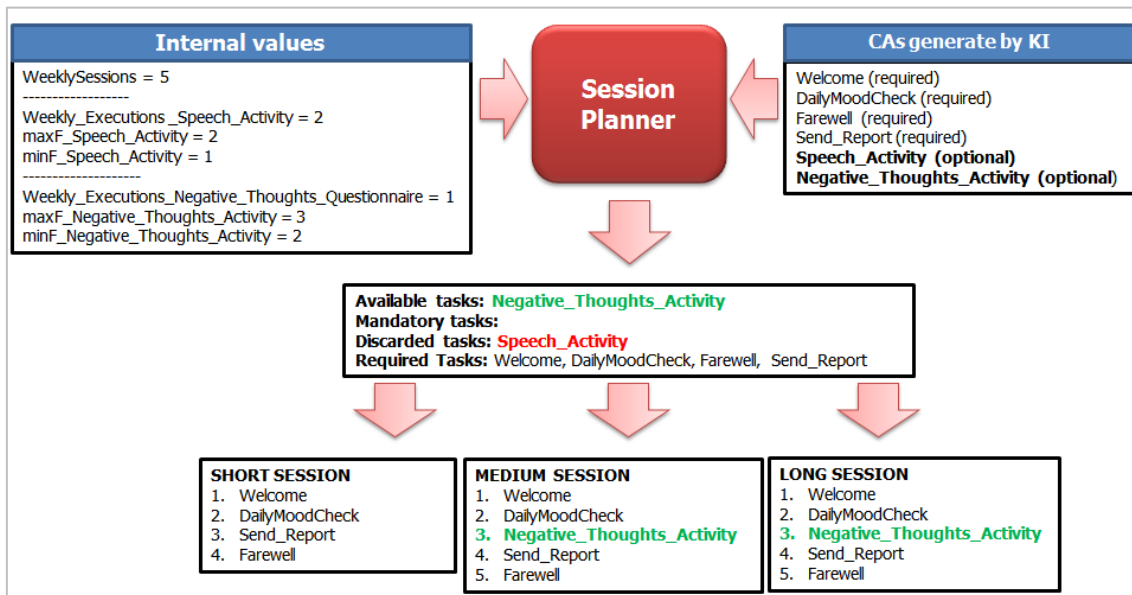

Figure 26: Example II of the SP Module execution.

Finally, in the last example (Figure 27) we consider that today is the last day of the week (there are 6 weeklySessions completed) and that the Negative_Thoughts_Activity has been performed four times in the week. Since its minimum frequency is set to five, this CA is labelled as mandatory. The SP must plan it in each one of the configuration sessions.
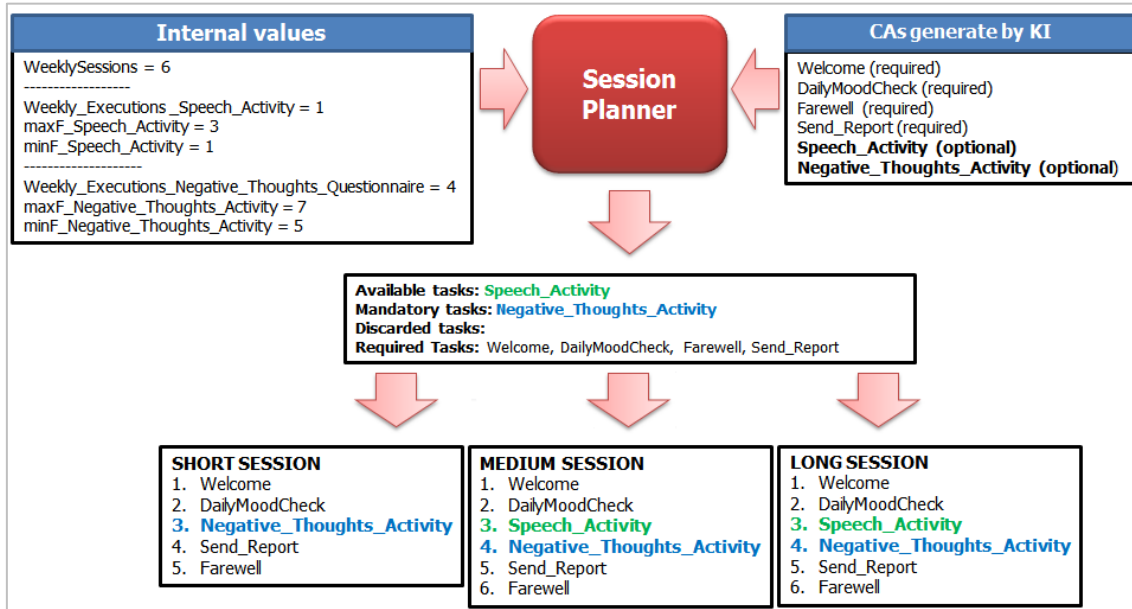
Figure 27: Example III of the SP Module execution.

4.3. The Event Generator Module (EG)

The EG is the mechanism responsible of the translation of the Planned Activity into a specific Event and it also generates the appropriate XML-based message to communicate with the CE module. Once the activities of the session have been planned in the SP, each activity is produced at different times during the session. The Planned Activity is passed to the event

generator module, which is responsible for formatting the specific activity as the Event that the CE module can appraise.

An example of the generated events in this module includes "Suicidal_Thoughts_Detected" which is inferred from the Crisis Plan activity (see Table 5). When the CE module receives this input, the appraisal process is performed and a XML-based message containing the specific *task*, and *emotion* to be executed by the VA during the next cycle of interaction is sent to the Dialogue Manager. The DM then implements the specific dialogue acts to generate the VA's phrases and display the corresponding facial expressions and/or head movements to represent the generated emotion. Similarly to the KE, KI and SP modules, the EG has been designed to easily implement further events that will produce new behaviours in the VA during the sessions.

Some examples about of parsing Planned Activities to Events are showing in the Table 5. Normally, this parsing is based on last KE evaluation.

Planned Activity	KE Assessment	Event
DailyMoodCheck_1 (code: 30000031)	Normal or null	Ready_For_DailyMoodCheck (code: 40000031)
	Good	Ready_For_DailyMoodCheck_Good_Assessment (code: 400000311)
	Bad	Ready_For_DailyMoodCheck_Bad_Assessment (code: 400000312)
DailyMoodCheck_2 (code: 30000032)	Normal or null	Ready_For_Second_Question_DailyMoodCheck (code: 40000032)
	Good	Ready_For_Second_Question_DailyMoodCheck_Good_Assessment (code: 400000321)
	Bad	Ready_For_Second_Question_DailyMoodCheck_Bad_Assessment (code: 400000322)
Crisis Plan (code: 30000091)	Any case	Suicidal_Thoughts_Detected (code: 400000912)
Introduce_PHQ9 (code: 30000200)	Normal or null	Ready_For_PHQ-9 (code: 40000200)
	Good	Ready_For_PHQ-9_Good_Assessment (code: 400002001)
	Bad	Ready_For_PHQ-9_Bad_Assessment (code: 400002002)

Table 5: Examples of Planned Activities parse to Events.

Although the whole EG functionality is not complex, it is important to maintain a correct parsing of the information between SP and EG (i.e. transform the planned activity into a specific event) facilitating the independence and modularity of each module. Moreover, the events generated by the EG can be reformatted to be adapted for a different GUI (e.g. in a different PHS which does not necessarily uses a Virtual Agent for the interaction with the patient) and display the relevant information about the disease treatment.

For the Help4Mood approach, when the SP has an empty plan and can't select a plannedActivity, it sends a NULL to the EG. When the EG receives a NULL value, it transforms this signal into a new Event that indicates that it has arrived at the end of the current session. This information will be used by the other modules on the Help4Mood system to e.g. execute all the necessary steps to finishes the execution of the system.

Chapter 5

Emotional Component

5.1. The Cognitive-Emotional Module (CE)

The Cognitive-Emotional module is the mechanism responsible of inferring an adequate and coherent emotion during the interaction with the user. This emotion is displayed by the VA with the aim to improve the VA-patient interaction, fostering the use of the system and encouraging patients to complete longer daily sessions.

The selected architecture to implement our cognitive-emotional module is **FAtiMA** (FearNot Affective Mind Architecture) [56]. FAtiMA is based on appraisal theory, implementing the **OCC model** [63] which defines a hierarchical organization of 22 emotion types (Figure 28). In OCC the appraisal process is carried out through the assessment of events that occur in the environment relating to the individual's *goals*, *standards* and *attitudes*. According to this model each event in the agent's environment is appraised according to the good or bad consequences of the event (for the own agent or for others); also if the detected event is a result of an action (performed by the self or others) and the aspect of the existent objects in the environment, then the corresponding emotion is generated.
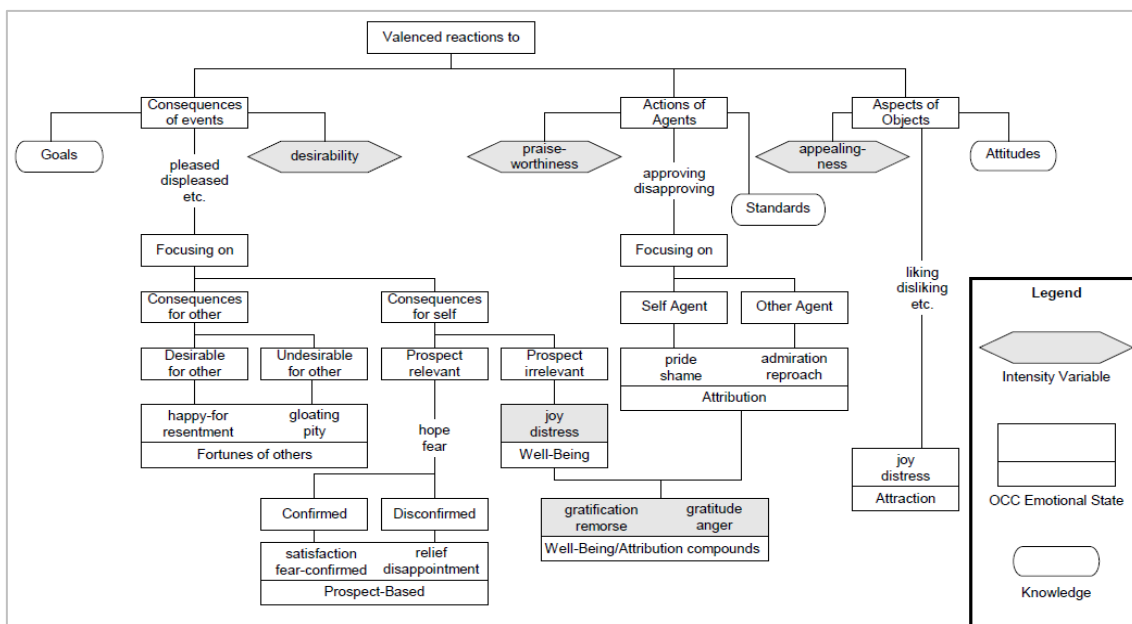

Figure 28: The OCC Model **[63]**

The implementation of the cognitive-emotional module in this Master Thesis [10] has been done through the development of a stand-alone Java application which makes use of the FAtiMA software libraries. The developed software includes mechanisms for handling and generating messages to interact with the rest of the Help4Mood components. Also, the design and implementation of the interface to get and set the required data in the FAtiMA architecture was developed.

The cognitive-emotional module (Figure 29) receives from the EG the specific event and uses a set of pre-defined **emotional reactions**, **action tendencies**, **goals** and **actions** in order to generate the agent's behaviour. All the VA emotions, emotional reactions, action tendencies and goals involved in the appraisal mechanism are defined in XML-based

configuration files to facilitate the addition of new values in the future that can produce longer and more complex daily sessions. All the events received from the EG are also XML-based messages containing the required information to perform the appraisal process. The result of this appraisal generates the next task that the VA will perform during the interaction with the user jointly with the corresponding emotion. This <task, emotion> pair is also formatted as an XML message and passed to the DM to execute the next cycle of interaction. The DM basically implements the dialogue corresponding to the task and forwards the emotion information to the graphical module of the VA to display the facial expression corresponding to the type and intensity of the generated emotion.
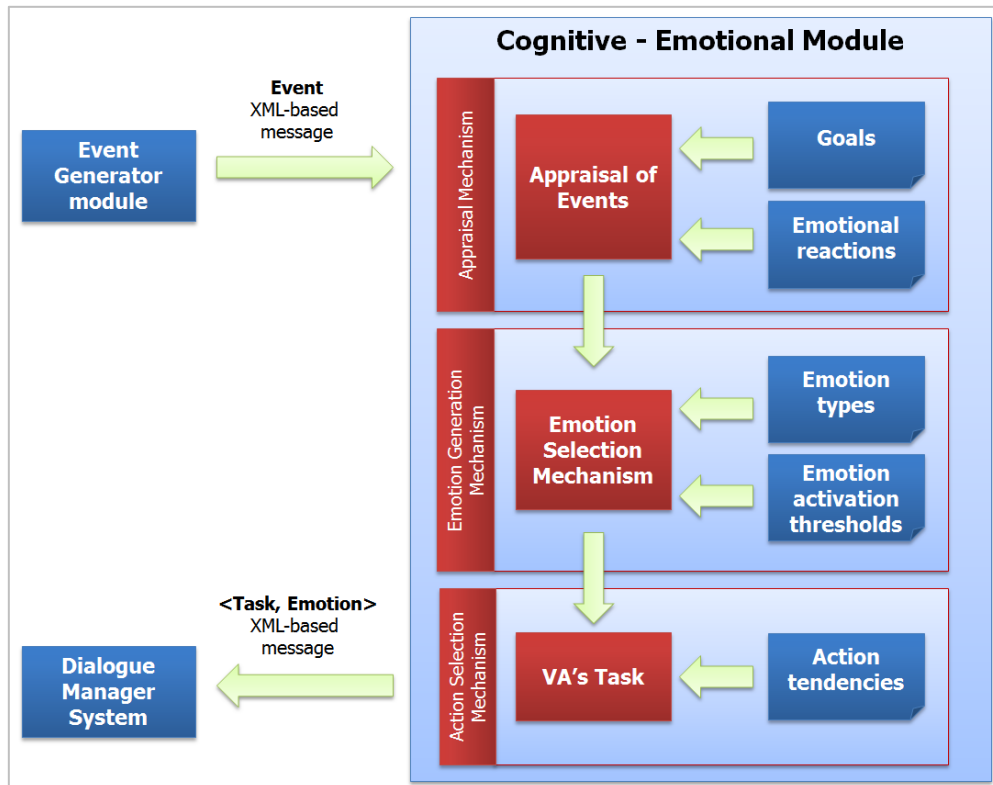
Figure 29: The Cognitive-Emotional module

Two different personalities have been modelled in the current version of the cognitive-emotional module to produce two different styles of interaction: a **friendly** and a **formal VA**. The positive emotions displayed by the friendly VA are more frequently shown than when interacting with a formal VA, which adopt a more emotionally neutral stance. This attitude in the VA's behaviour style is achieved by defining different thresholds for the triggering of the emotions, as well as different values for the emotional reactions, goals, and action tendencies. Different values in these parameters produce different appraisal results even on the same event, which in turn produce a different emotion, or different emotion intensity, in the VA.

Taking into account the special characteristics of the Help4Mood users, the clinicians of the consortium recommended some kind of regulation in the empathic responses conveyed by the VA. In this sense, the range of emotions produced in the VA should be restricted to neutral or positive emotions, avoiding the displaying of negative emotions that can be interpreted as sympathetic expressions of condolence that may imply a sense of unintended agreement with the patient's (negative) views [64] [65]. In consequence, the system should not produce empathic behaviours by adopting the same –typically negative– emotions when events are appraised as non-desirable (e.g. when the patient is reporting negative moods and thoughts, or when something bad in the state of the patient is inferred from the data collected). Following this recommendation the EC module implements a basic model of therapeutic empathy where all the negative emotions are simply suppressed for not to display them in the VA during the interaction with the patient. This kind of emotion regulation is what Gross process model of

emotion regulation defines as *response modulation* where the consequences (experiential, behavioural and physiological) of the (in our case negative) emotions are suppressed [66].

Therefore, in the current implementation, we have selected only three positive emotions from the OCC model to be used (including a **neutral** stance –i.e. no emotion–): **joy**, **happy-for**, and **admiration**, as described as follows:

- **Joy**: activated when an event is appraised as desirable for the VA. Examples of these events include the daily logging of the user into the system (representing the achievement of one of the ultimate goals of the VA: maintain a good level of adherence to the treatment), or the user's acceptance of the activities offered by the VA during the session. This positive aspect will be communicated to the user through some specific words in the interaction dialogue such as "Thanks for logging in today, I am happy to see you again" or "Great! I'm glad that you accept the plan for today's session".
- **Happy-for**: elicited when an event is appraised as desirable for the user. Examples include good self-reported moods or thoughts and the detection of any improvement in the patient's condition through inferences from the data collected. Again, examples of positive utterances in this context are "That's good, I'm happy to hear that" or "It seems that you are achieving your targets. Good for you" respectively.
- **Admiration**: activated when an event is appraised as a desirable consequence of a user's action. Examples include the correct completion of the proposed activities during the session or the completion of the whole session and the confirmation (by self-reporting) that the user has followed the recommendations provided by the VA (e.g. to improve physical or sleep activities). An appropriate positive response would be as "You are doing well!".

Emotions thresholds and decay rates were defined in the FAtiMA configuration files. Joy, Happy-for, and Admiration emotions were defined with low threshold. High values in the emotion thresholds prevent the activation of the corresponding emotion (this was done for the rest of the 19 OCC emotion types including all those negative emotions). In the cases when no emotion is activated, the VA adopts the default neutral emotional behaviour.

```
<Character role="FriendlyVA">
<EmotionalThresholds>
        <EmotionalThreshold emotion="Joy" threshold="1" decay="5"/>
        <EmotionalThreshold emotion="Happy_For" threshold="1" decay="7"/>
        <EmotionalThreshold emotion="Admiration" threshold="1" decay="8"/>
        <EmotionalThreshold emotion="Love" threshold="9" decay="2"/>
        <EmotionalThreshold emotion="Hate" threshold="9" decay="2"/>
        ...
</EmotionalThresholds>
...
</Character>
```

Figure 30: Emotions and thresholds definitions

With the set of emotions defined, we also authored the **emotional reactions** and **action tendencies** that the VA can produce. The emotional reactions represent how the VA appraises the events produced in the environment (in Help4Mood, these events are generated by the EG module). Conversely, action tendencies represent the VA's reactions to particular emotional states elicited in the VA. Based on the currently identified session's activities, a set of emotional reactions rules and action tendencies were authored in FAtiMA. Some examples of the emotional reaction rules defined for the friendly VA are the following:

```
<Character role="FriendlyVA">
<EventReactions>
        <EmotionalReaction desirability="8" praiseworthiness="3">
                <Event subject="User" action="Open_Session"/>
        </EmotionalReaction>
        <EmotionalReaction desirability="5" praiseworthiness="2">
                <Event subject="User" action="Ready_For_DailyMoodCheck"/>
        </EmotionalReaction>
        <EmotionalReaction desirability="8" desirabilityForOther="3" praiseworthiness="2">
                <Event subject="User" action="Ready_For_DailyMoodCheck_Good_Assessment"/>
```

```
            </EmotionalReaction>
            <EmotionalReaction desirability= "-3" desirabilityForOther= "-7" praiseworthiness= "2">
                    <Event subject= "User" action= "Ready_For_DailyMoodCheck_Bad_Assessment"/>
            </EmotionalReaction>
            <EmotionalReaction desirability= "-3" desirabilityForOther= "-7" praiseworthiness= "2">
                    <Event subject= "User" action= "Ready_For_Select_Session_Type_Bad_Assessment"/>
            </EmotionalReaction>
                    <EmotionalReaction desirability= "3" desirabilityForOther= "7" praiseworthiness= "2">
            <Event subject= "User" action= "Ready_For_Select_Session_Type_Good_Assessment"/>
            </EmotionalReaction>
            ...
</EventReactions>
<ActionTendencies>
        <ActionTendency action= "Congratulate(User)">
                <Preconditions>
                </Preconditions>
                <ElicitingEmotion type= "Happy_For" minIntensity= "5">
                <CauseEvent subject= "*" action= "HighScorePHQ9" target= "User"/>
                </ElicitingEmotion>
        </ActionTendency>
        <ActionTendency action= "look_at_content(User)">
                <Preconditions>
                </Preconditions>
                <ElicitingEmotion type= "Happy-For" minIntensity= "2">
                <CauseEvent subject= "User" action= "Ready_For_DailyMoodCheck" target= "*"/>
                </ElicitingEmotion>
        </ActionTendency>
        ...
</ActionTendencies>
...
</Character>
```

Figure 31: Emotional reactions and action tendencies definition

Based on the OCC model, FAtiMA uses the desirability variable to elicit the well-being emotions (see Figure 28): a positive value produces the joy emotion and the distress emotion is produced otherwise. Additionally, the value in the praiseworthiness variable is used to generate the attribution emotions. Depending on the subject responsible for the event (the self or other agent) and on the positive or negative value of the variable, one of the following emotions can be elicited: pride (if subject="self-agent" and praiseworthiness > 0); shame (if subject="self-agent" and praiseworthiness < 0); admiration (if subject="other agent" and praiseworthiness > 0); and reproach (if subject="other agent" and praiseworthiness < 0). Finally, the desirabilityForOther variable is used to generate the fortune-of-others emotions: happy-for, pity, gloating and resentment.

Not all the VA's actions should necessarily be reactive responses linked to a specific emotion. There are some actions the can be executed according to goals that the VA might have. For this case, FAtiMA offers a mechanism which uses a set of defined goals in the VA to fire the corresponding action(s) to achieve that goal(s). Examples of the authored goals for the Help4Mood VA are the following:

```
<Character role= "FriendlyVA">
<Goals>
    <Goal name= "Welcome(User)" importanceOfSuccess= "2" importanceOfFailure= "2"/>
    <Goal name= "Welcome_Reminder(User)" importanceOfSuccess= "2" importanceOfFailure= "2"/>
    <Goal name= "DailyMoodCheck_1(User)" importanceOfSuccess= "2" importanceOfFailure= "2"/>
    <Goal name= "DailyMoodCheck_1_Good_Assessment(User)" importanceOfSuccess= "2"
importanceOfFailure= "2"/>
    <Goal name= "DailyMoodCheck_1_Bad_Assessment(User)" importanceOfSuccess= "2"
importanceOfFailure= "2"/>
    <Goal name= "Select_Session_Type(User)" importanceOfSuccess= "2" importanceOfFailure= "2"/>
    <Goal name= "Select_Session_Type_Good_Assessment(User)" importanceOfSuccess= "2"
importanceOfFailure= "2"/>
    <Goal name= "Select_Session_Type_Bad_Assessment(User)" importanceOfSuccess= "2"
```

```
importanceOfFailure= "2"/>
    <Goal name= "Sleep_Questionnaire(User)" importanceOfSuccess= "2" importanceOfFailure= "2"/>
    <Goal name= "Sleep_Questionnaire_Good_Assessment(User)" importanceOfSuccess= "2"
importanceOfFailure= "2"/>
    <Goal name= "Sleep_Questionnaire_Bad_Assessment(User)" importanceOfSuccess= "2"
importanceOfFailure= "2"/>
    <Goal name= "Activity_Monitoring(User)" importanceOfSuccess= "2" importanceOfFailure= "2"/>
    <Goal name= "Activity_Monitoring_Good_Assessment(User)" importanceOfSuccess= "2"
importanceOfFailure= "2"/>
    …
</Goals>…

</Character>
```

Figure 32: Goals definition

In order to clarify the process of the CE module, we show a full example about the **Daily Mood Check activity**. This questionnaire starts when the cognitive-emotional module receives the event that indicates the user is ready to start the questionnaire. This is illustrated in the XML representation below:

```
<?xml version= "1.0" encoding= "UTF-8" standalone= "yes"?>
<ns2:Event xsi:schemaLocation= "urn:event.data.eg.bmg.ibime.upv.es ./docs/event.xsd"
xmlns:ns2= "urn:event.data.eg.dss.h4m.bmg.ibime.upv.es"
xmlns:xsi= "http://www.w3.org/2001/XMLSchema-instance">
<typeOfEvent>3</typeOfEvent>
<description>
        <code>40000031</code>
        <snomed>NO</snomed>
        <name>Ready_For_DailyMoodCheck</name>
</description>
<generated>2012-08-21T12:50:40.783+02:00</generated>
</ns2:Event>
```

Figure 33: Example of Event XML message

The received event is appraised and the authored goal generates the task to ask the first question of the daily mood check questionnaire.

```
<!— Goal to produce the first question of Daily Mood Check questionnaire -->
<ActivePursuitGoal name= "DailyMoodCheck_1(User)">
<PreConditions>
<RecentEvent subject= "User" action= "Ready_For_DailyMoodCheck"/>
</PreConditions>
<SucessConditions>
<RecentEvent occurred= "True" subject= "[SELF]" action= "First_DMC_Question" target= "User"/>
</SucessConditions>
</ActivePursuitGoal>
```

Figure 34: DailyMoodCheck_1 related Goal

Once the user answers the question through the GUI, the next received event from the EG reports this user action to initiate the next question. The event generated by the EG can also indicate that the user's response has been assessed as positive. A positive assessment of the user's response produces the Happy-for positive emotion in the friendly version of the VA which is displayed through a smile and the use of a specific utterance before asking the next question (e.g. "That's great!" "Now the next question is…").

```
<!—Emotional reaction to a good result in the first question of the Daily Mood Check-->
<EmotionalReaction desirability= "8" desirabilityForOther= "10" praiseworthiness= "2">
        <Event subject= "User" action= "Good_Result_First_Question_DMC"/>
</EmotionalReaction>
```

Figure 35: Emotional reaction example

This process is executed after each question until the complete questionnaire is administered. The emotional behaviour of the VA during the administration of the questionnaire is also modelled different for the two personality types. The positive happy-for emotion is triggered by the friendly VA each time a positive response is assessed after every question, and

in consequence a positive feedback is provided to the user. In the formal VA, the positive emotion is triggered only when the global score of the whole questionnaire reaches a predefined threshold. In this case, the positive feedback provided to the patient appears at the end of the questionnaire.

The final xml output message from EC module contains the pair <task, emotion> which is passed to the Dialogue Manager component of the VA to generate the corresponding utterances and non-verbal behaviour that is conveyed during the interaction with the patient. An example of this <task, emotion> message is as follows:

```xml
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<ns2:TaskEmotion
xmlns:ns2="urn:TaskEmotion.data.MyObserverComponent.FAtiMA.ce.h4m.bmg.ibime.upv.es"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="urn:TaskEmotion.data.MyObserverComponent.FAtiMA.ce.h4m.bmg.ibime.upv.es
TaskEmotion.xsd">
<Task>
        <typeOfTask>5</typeOfTask> // CBT interaction
        <description>
                <code>30000031</code>
                <snomed>NO</snomed>
                <name> DailyMoodCheck_1</name>
        <description>
        <start>2013-01-02T18:15:01.0Z</start>
        <end></end>
        <parameters/>
        <cause>
                <code>40000011</code>
                <timestamp>2013-01-02T18:14:25.0Z</timestamp>
        </cause>
</Task>
<Emotion>
        <typeOfEmotion>HAPPY-FOR</typeOfEmotion>
        <valence>1</valence>
        <intensity>3.2586</intensity>
        <decay>20</decay>
        <facialExpression>look_at_content(User)</facialExpression>
        <cause>
                <EventCause>
                        <code>40000011</code>
                        <timestamp>2013-01-02T18:14:25.0Z</timestamp>
                </EventCause>
                <TaskCause>
                        <code>30000011</code>
                        <description>Welcome</description>
                </TaskCause>
        </cause>
</Emotion>
</ns2:TaskEmotion>
```

Figure 36: Example of Task-Emotion pair XML message

Chapter 6

Patient Progress Report

The clinical summary reports support the healthcare decisions about the treatment process with information about the progress of a patient.

The patient progress report summarises the data to assess the evolution of the patient from the data collected during the daily sessions. This report highlights the main trends in the patient's mood, behaviour, cognitive and psychomotor function. Following a user-centred design methodology, the recommendations and preferences from the clinicians of the Help4Mood consortium have been continuously considered during the development of the report. This summary report has been defined as an A4 PDF sheet where some graphics and text are used to present the information from the patient in order to get valuable and easily-digestible form to make the assessments on patient's condition. The report includes the progress of the patient (e.g. in mood and activity) during the current week in comparison with the past four weeks. One of agreed objectives of the report is that the patient and his/her clinician make a joint review of the document and facilitate the discussion about the individual's progress, the experience of depression and path to recovery.

The progress report can be analysed by both, patients and clinicians. One of the tasks implemented in the PHS's sessions offers to the patient (every week) the option to visualise this report in his/her own laptop. Additionally, this report can be send to his/her clinician. Different strategies to send the report between to the clinical site have been implemented. We can set the sending method in the configuration files of the system. The system can use a list of emails defined in the configuration files in order to send this PDF report by email directly to the clinicians. Other possibility is the sending of the report to the Health Information system of the clinical centre using HL7-CDA in order to integrate this information into the particular health information system. This kind of integration is currently available in one of the clinical institutions of the consortium: the Sant Joan de Deu - Serveis de Salut Mental (SJD-SSM) in Barcelona.

In the following sections we describe the process followed to summarise the relevant clinical information and the work performed to get the visual appearance of the final summary report in the PDF file.

6.1. Visual Appearance

The design of the visual appearance was performed following the recommendations of the Microsoft Health Common User Interface (MSCUI [27]) guides. MSCUI is a guidance to design effective and user-friendly clinical data interfaces that maximise patient safety and increase clinical effectiveness. Although it is not a standardised methodology, it is supported and defined by the Microsoft Health group (clinicians and technicians) getting a wider acceptance and international prestige.

From MSCUI, we follow the recommendations concerning the identification of the patient (the patient must be identified in the header, using full name and ID), the representation of the graphs and tables (such as the label identification format, X-Axis and Y - Axis Arrangements, the use of gridlines and the correct typeface formatting)

In the Figure 37 we can see an example of the report template with all the information contained in the PDF file. This information is organized as follows:

---

[27] Microsoft Health Common User Interface guidance overview: http://www.mscui.net/DesignGuide/DesignGuide.aspx

- **Header**: In this area the report displays the information of the period date, and the name and identification of the patient
- **Overall Progress**: This text is generated by the Natural Language Processing component of the Help4Mood system. It summarises textually the data collected with the Daily Mood Check and the Negative Thoughts Questionnaires.
- **Summary Plot**: This easy-to-understand plot is generated by the Data Analysis module (see section 4.1.1). It presents the progress of the sleep quality and the amount of activity detected in the patient. It also detects changes in daily physical activity patterns to try to contribute to a better assessment of the progress about the patient's wellbeing.
- **Session Statistics**: In this area the report shows important information about the scores of the PHQ9, and best and worst results regarding the data from the sensor devices.
- **Thoughts:** This table summarizes the thoughts selected by the patient in the negative thoughts questionnaire.
- **Mood Records**: This area contains two plots. The first one displays the scores of the Daily Mood Check questionnaire. The second one presents the same information but grouped in a period of a week. Both plots are generated by the DA module (see section 4.1.1).
- **Activity Record**: These plots are very useful in order to known the adherence and the reliability of the gathering data from the PMS. These plots show (daily and weekly) the sleep and activity periods, and the no data period. Both plots are also generated by the DA module (see section 4.1.1).
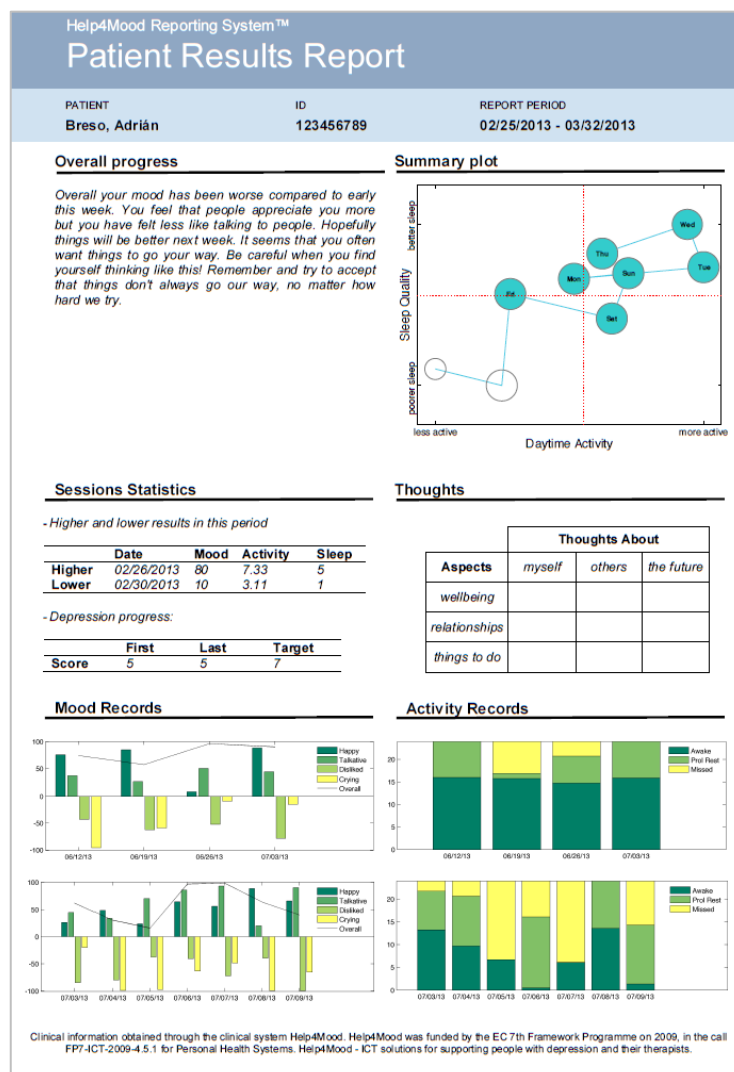
Figure 37: Example of progress patient report

48

6.2. **Clinical Summarization**

Clinical summarisation is the process by which relevant information is electronically summarised and presented at the point of care. Feblowitz et al [60] proposed the first conceptual model (AORTIS) for describing and understanding clinical summarisation. Laxmisan et al. [67] used AORTIS model in order to compare the general clinical summary screens of different Electronic Health Record System (EHRs) such as Open MRS, Epic, or Star Panel.

The AORTIS model identifies five distinct stages of clinical summarisation:

1. **Aggregation**: is the collection of clinical data from various available sources.
2. **Organisation**: is the structuring of data according to some underlying principle without condensing, altering, or interpreting it. Common organisation operations are grouping (e.g. by data type or origin of service), sorting (e.g. by date or alphabetically), and periodization (e.g. by urgency or specialty).
3. **Reduction and/or Transformation**: reduction is the process of filtering salient information from the database without altering it to decrease the amount of data presented (e.g. show only the most recent values or the data from a single location). Transformation is the process of altering how the data is viewed or how the data is presented in order to facilitate understanding (e.g. statistical medians or percentiles).
4. **Interpretation**: Is the application of the general medical knowledge to a single data type in order to identify abnormally high or low results.
5. **Synthesis**: is the combination of two or more data elements along with knowledge-based interpretation of patient state to yield meaning or suggest action.

Following the AORTIS model of Feblowitz [60], se implemented the patient report functionality in the DASPS (the Manager Report Generator module) taking into account the different proposed strategies (see Figure 39):

1. **Aggregation:** We collect clinical data from different sources, all of them nurtured at different point of times during the interactive sessions: (1) from the KE module, (2) from the KI module, (3) from the DA module, and (4) from the DM component. All these data are stored into the jessData.xml persistent file.
2. **Organization:** We organize these data grouping them by date and session.
3. **Reduction and/or Transformation:** We filter the original information by a selected period (normally, last four weeks). Additionally, the DA module transforms the data of the PMS into valuable and understandable information (e.g. higher and lower values, means, etc.) and stores it into the jessData.xml file.
4. **Interpretation:** The interpretation of the information is carry out in DA module. The clinicians of the consortium spent a lot of time discussing about what would be the best way to graphically represent the data of the patient and to detect normal or abnormal behaviour. This clinical knowledge was implemented into the DA module in order to generate all the plots of the report including the Summary the Mood and the Activity plots.
5. In this stage, we include the definition of the report template. Clinicians agreed on a format for the visualization of information. This report must be a single sheet with the appearance and the information presented in Figure 37. We used the JasperReport library[28] in order to define de template and generate the final version of the report.
6. **Synthesis:** In briefInfo.xml file (Figure 38) we combine all the information which will be added to the report. This file is sending to the JasperReport Library and parsing it using the defined template of the report to generate the final PDF document. Finally, this PDF report can be sent to the clinicians and also visualised by the patient.

---

[28] http://community.jaspersoft.com/project/jasperreports-library

```xml
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<ns2:Briefinfo xmlns:ns2="urn:briefinfo.data.report.h4m.bmg.ibime.upv.es"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="urn:briefinfo.data.report.h4m.bmg.ibime.upv.es ./config/config-dss/docs/briefinfo.xsd">
    <h4m_Data>
        <resources>
            <headerImage>./config/config-dss/report/logoH4M.jpg</headerImage>
            <chart_Week_Mood>./config/config-dss/report/WeekMood.tif</chart_Week_Mood>
            <chart_Week_Activity>./config/config-dss/report/WeekActivity.tif</chart_Week_Activity>
            <chart_Week_Sleep>./config/config-dss/report/WeekSleep.tif</chart_Week_Sleep>
            <chart_Daily_Mood>./config/config-dss/report/DailyMood.tif</chart_Daily_Mood>
            <chart_Daily_Activity>./config/config-dss/report/DailyActivity.tif</chart_Daily_Activity>
            <chart_Daily_Sleep>./config/config-dss/report/DailySleep.tif</chart_Daily_Sleep>
            <chart_Thoughts>./config/config-dss/report/NoData.bmp</chart_Thoughts>
        </resources>
        <patient>
            <id>123456789</id>
            <name>Adrian Breso</name>
        </patient>
        <lastLogin>2013-03-14T10:02:45+01:00</lastLogin>
        <logins>6</logins>
        <finishedSessions>0</finishedSessions>
        <elapsedTime>8750</elapsedTime>
        <reportFrom>2013-02-14T16:26:11.968+01:00</reportFrom>
        <reportTo>2013-03-14T16:26:11.968+01:00</reportTo>
        <text1>text1...blablabla</text1>
        <text2>text2...blablabla</text2>
        <text3>text3...blablabla</text3>
        <valueBestMood>1</valueBestMood>
        <valueBestActivity>0</valueBestActivity>
        <valueBestSleep>0</valueBestSleep>
        <dateBestValues>2014-02-03T00:00:00+01:00</dateBestValues>
        <valueWorstMood>1</valueWorstMood>
        <valueWorstActivity>0</valueWorstActivity>
        <valueWorstSleep>0</valueWorstSleep>
        <dateWorstValues>2014-02-03T00:00:00+01:00</dateWorstValues>
        <valueFirstDepressionScore>9</valueFirstDepressionScore>
        <dateFirstDepressionScore>2013-03-14T10:00:10+01:00</dateFirstDepressionScore>
        <valueRecentDepressionScore>9</valueRecentDepressionScore>
        <dateRecentDepressionScore>2013-03-14T09:35:01+01:00</dateRecentDepressionScore>
        <valueTargetDepressionScore>NULL</valueTargetDepressionScore>
    </h4m_Data>
</ns2:Briefinfo>
```

Figure 38: Example of the briefInfo.xml file
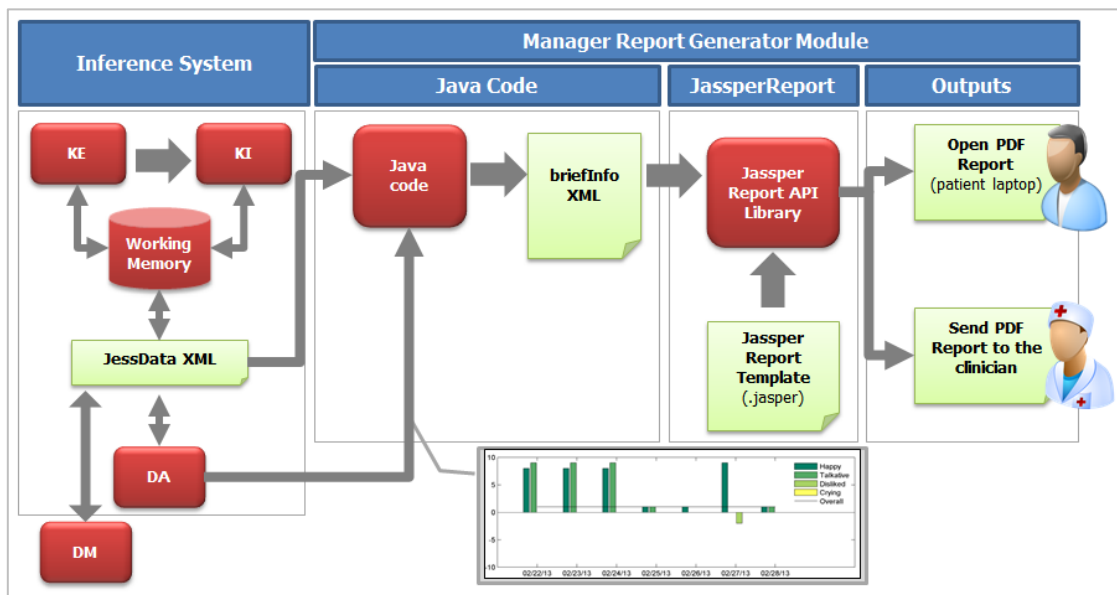
Figure 39: Pipeline of the patient report generation

Chapter 7

Evaluation

7.1.   Internal Tests

In order to test the correct functionality of the developed rules in the KE and KI modules, a JAVA standalone application previously developed in the IBIME group was adapted for assessing the efficiency and effectiveness of the implementation. It provides automatic data generation and combination, Jess rules execution, multi tabular results view and execution time measuring. This tester tool was very usefully in order to fix rule execution problems (mainly syntactic problems). Figure 40 shows the tester in the prototype implementation of the Help4Mood Data Processing and Analysis Module.
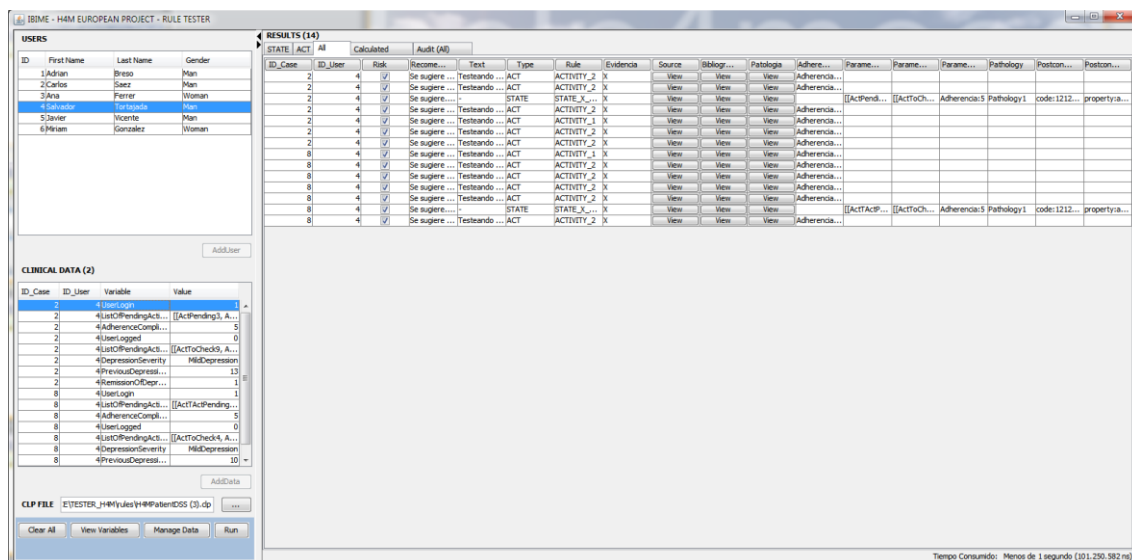
Figure 40: Java standalone tester tool

We also developed an additional Java standalone application to visualise and manage the content of the jessData.xml file (Figure 41 and Figure 42) which is encrypted due to medical legal restrictions. This tool (called jessDataViewer) allowed us to test if the data are generated and stored correctly during the execution of the patient's sessions.

Figure 41: JessDataViewer Java standalone application (table view)

Figure 42: JessDataViewer Java standalone application (tree view)

An additional Java application was also developed to simulate the exchange of information between the DASPS module with the rest of the components of the Help4Mood system. This tool, called GUI_Simulator.java (Figure 43), allow us the testing of the complete functionality of the DASPS module by simulating the inputs and outputs of the PMS and the DM components of the full PHS facilitating all the internal tests without a full system integration.

We simulated the activity data received from the PMS at the beginning of the daily session and we generate the initial message to be sent to the DM: the LogIn_Message. With this initial message the system (i.e. a new session) starts. The GUI_Simulator shows us the System Inference inferences, the classification of the CAs by the SP, the past planning, the provider planning, the current PA, the KE assessment, and the final message that the EG sends to the DM (which contains the task and the emotion to be executed by the Virtual Agent).

After this execution, we can simulate the response of the DM in order to make a realistic interaction between the components like a real daily interactive session with the patient.
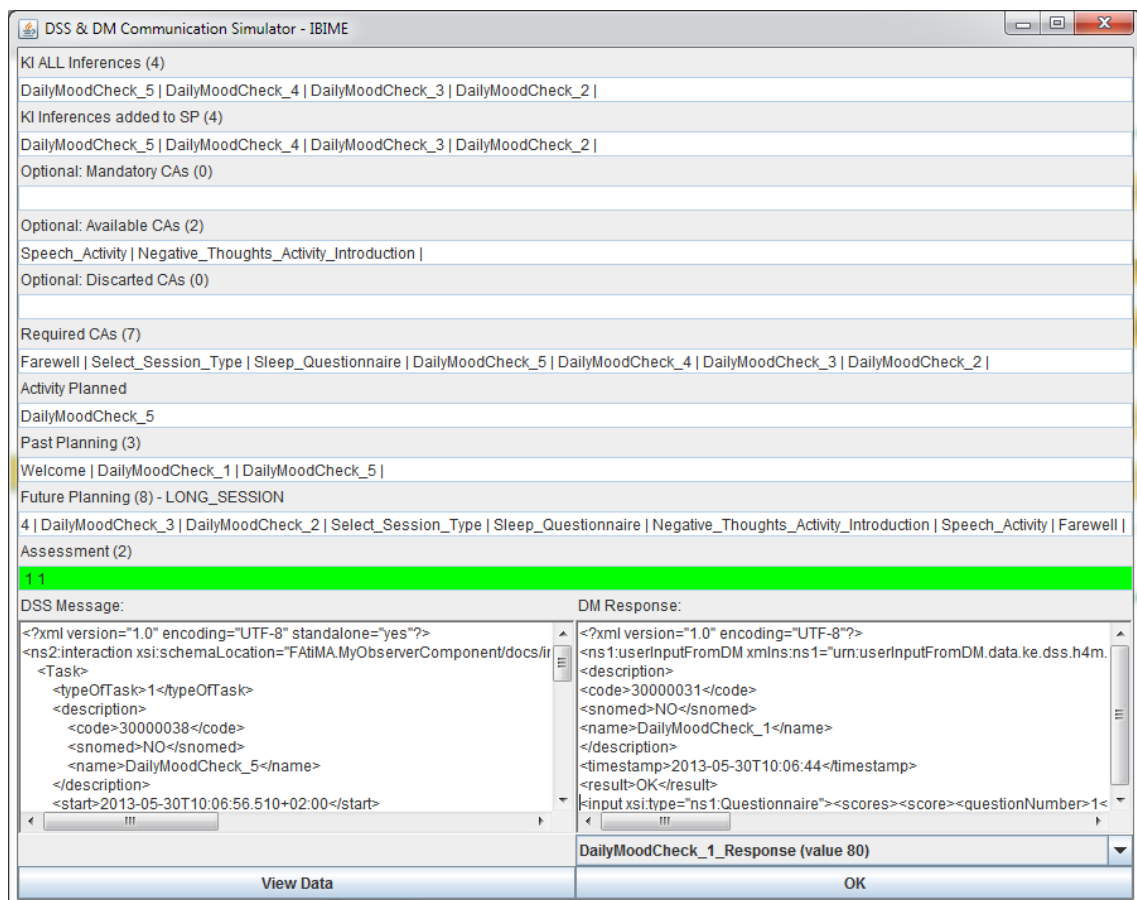
Figure 43: Screenshot of GUI_Simulator.java

Additionally we can visualise (and modify) the current state of the jessData.xml to test the correct functionality of the storage system (Figure 41 and Figure 42).

The results obtained with the GUI_Simulator were the expected results. All implemented rules and the rest of the modules implementation worked properly. So these results met the expectations of the clinicians and technicians of the project.

7.2. **External Tests**

For the evaluation of the whole system, the Help4Mood consortium planned to carry out three real pilots to obtain feedbacks from patients and clinicians (see Table 6). For these pilots, the

consortium provides to a set of selected patients with a laptop with the Help4Mood system, a set of sensors devices and a questionnaire that the clinician administered during an exit interview to collect the feedback from the patient about the system.

Functionalities	Prototype Stage		
	First	Intermediate	Final
Welcome	x	x	x
Daily Mood Check	x	x	x
Actigraphy	x	x	x
Closing Summary	x	x	x
PHQ-9		x	x
Negative Thought Questionnaire		x	x
Speech Data		x	x
Crisis Plan		x	x
Summary Report		x	x
Sleep Questionnaire		x	x
Behaviour Activation			x
Relaxation			x
Attentional Bias Modification			(x)

Table 6: Planning of the three Help4Mood pilots

Currently, the consortium only carried out the first pilot but the current state of the architecture described in this Master Thesis is able to carry out the complete functionality for the second pilot.

The first pilot involved 13 participants (see Table 7) using the system during 7 days. Participants were aged between 18 and 64 inclusive. They had a previous episode of Major Depression Disorder diagnosed by a general practitioner, psychiatrist or clinical psychologist between 6 and 18 months previously, but were either fully recovered or had only residual mild depressive symptoms.

Country	Participants		
	Male	Female	All
Scotland	2	0	2
Romania	4	3	7
Catalonia	2	2	4

Table 7: Participants distribution in the first prototype.

The first prototype was the simplest of the three planned pilots. Only four tasks were provided to the patient through the VA (see Table 6), so the duration of the daily sessions were very short. This causes that the whole functionality of the SP module was not full exploited at this stage (not much tasks for planning during a session). Moreover, the period of data collection was also short and the DA module could not get important conclusions about patient's activity patterns. In general, during this pilot the system described in this work was operated below its full potential.

The conclusions obtained from the first pilot related with the architecture described in this Master Thesis are summarized as follows:

- The perception of the emotional responses in the VA was divided in the participants of the pilot. While some of them notice the positive reactions in the VA, some other state that the emotional behaviour of the VA was flat.

- For clinicians, patient-adapted displays and summaries of Help4Mood data can be important tools for structuring and optimally using the limited time that clinicians

get to spend with patients. In this way, far from replacing or undermining the therapeutic relationship between patient and clinician, Help4Mood can actually strengthen it.

- Help4Mood can provide a systematic framework for data collection within clinical studies and trials.

- Help4Mood can provide a data representation that future evidence-based rules can build on.

Chapter 8

Conclusions & Future work

The original objectives of the work proposed in this Master Thesis have been met in the following way:

- We developed a modular, standard, and flexible DASPS based on established standards of communication and data representation (communication: XML; clinical terms: SNOMED CT). In this sense, our DASPS can be adapted to other clinical scenario in which Remote Patient Monitoring and the delivery of daily sessions are important aspects of the treatment (e.g. other mental disorders). The developed DASPS component contained a deliberative component: Inference System (DA+KI+KE), Session Planner, and Event-Generator; and an emotional component: Cognitive-Emotional module.

  - By means of the DA module, we analysed the data collected from the PMS to detect changes in the patterns of physical activity of the patient. We can identify these changes by representing the analysed data in a plot (see Figure 11).

  - We developed the KI and KE modules using the JESS library and expert clinical knowledge from clinicians of the Help4Mood consortium. Our Inference System is able to generate clinical tasks and clinical concepts (using SNOMED CT terminology or internal codification). Currently there are implemented more than 40 clinical tasks to be used during the VA-patient interactive sessions (see Annex A).

  - We implemented a customizable, adaptive, and flexible Session Planner module (SP) that generates a personalised patient-based daily session. This daily session is updated during the session according to the patient answers and the clinician preferences facilitating the adherence of the patient to the treatment.

  - The system is able to gather data related to the patient's mood by offering activities based on Cognitive Behaviour Therapy (CBT), such as the innovative Negative Thoughts Questionnaire (NTQ), psychomotor speech tasks, and Daily Mood Check (based on CESD4)..

  - We stored all the important information generated during the sessions in order to prepare a weekly clinical progress patient report (see Figure 37). The report can be visualised by the patients and send it to the clinicians.

  - In order to improve and promote the patient adherence and the patient acceptance of the system, (1) the Cognitive-Emotional module adds adequate emotional responses during the interaction with the patient, (2) the Session Planner module adapts the tasks and the duration of the session on the fly according with the patient stamina, and (3) we evaluated the system in a real pilot in order to obtain patient and clinical feedback which we use for the further improvement of the system.

- We implemented a Cognitive-Emotional Module which has been the initial step towards the construction of a therapeutic empathy model [68] aimed to better adapt the emotional responses in the VA to the patient's inputs.

For the future, we are planning several developments in order to improve and evolve the current functionality of the DASPS. These improvements will be designed, implemented and evaluated as part of my future PhD work. The main proposed improvements are the following:

- Complement the evaluation of the system with the additional two real pilots (see Table 6). More users and more clinical tasks will be used to better assess if the developed system improves traditional treatment of Major Depression.

- Carry out a study of the correlation between physical activity and the patient's clinical condition.

- Extend the clinical summary report to include the relevant information extracted from the new added clinical activities to the daily sessions.

- Extend the Cognitive-Emotional module, to model more realistic therapeutic empathy responses in the VA. In this sense, the addition of some negative emotions is proposed but performing a different strategy to regulate these emotions: we propose to implement the *cognitive change* strategy of emotion regulation defined in [66]. This can be done through the extension of the FAtiMA architecture by adding a new *re-appraisal* component responsible to select the adequate type and intensity of the emotion during the interaction with the patient.

- Extend the current list of clinical concepts and clinical tasks provided by the DASPS. Three clinical tasks are currently identified to be added:

  o Behavioural Activation: Based on a pre-defined list of patient's preferred activities (such as go walking, cycling, watching a movie, etc.), during a daily session the VA should ask the patients which of these activities they want to do today, and motivate the patients to perform it. During the next session, the VA should ask whether the patient has carried out the selected activity and how he/she feels after that. The continuous suggestions of different *agreeable* activities would help in the behavioural activation of the patient and benefit the treatment [69].

  o Additional standardised questionnaires: New questionnaires to gather more information from the patient could be added. This includes specific questions to know e.g. the nutrition habits of the patient or questions related with medication intake.

  o Relaxation exercises. Short exercises may be defined by the clinicians to help in the relaxation of the patient either as a response to a particular event or to be performed in a routine way during a daily session.

  o Attention Bias Modification: Several studies showed that clinically depressed patients selectively attend to some kind of expressions. This bias is usually to sad, but not angry or happy faces [70]. The attention bias for sad stimuli was also found in remitted Major Depression, as well as following sad mood induction in at-risk populations (i.e., children of depressed mothers). Preliminary evidence suggests that attention biases to sad words may correlate with symptom severity [71].

Bibliography

[1] K. Jamison, "The many stigmas of mental illness," *Lancet,* vol. 367, pp. 533-534, 2006.

[2] N. Kennedy and K. Foy, "The impact of residual symptoms on outcome of major depression," *Curr Psychiatry Rep.,* vol. 7, pp. 441-446, 2005.

[3] P. Sobocki, B. Jönsson, J. Angst and C. Rehnberg, "Cost of depression in Europe," *J Ment Health Policy Econ,* vol. 9, no. 2, pp. 87-98, 2008.

[4] M. Wolters, C. Burton, C. Matheson, A. Bresó, A. Szentagotai, J. Martinez-Miranda, E. Fuster, J. Rosell, C. Pagliari and B. McKinstry, "Help4Mood - Supporting Joint Sense Making in the Treatmet of Major Depressive Disorder," in *Workshop on Interactive Systems in Healthcare (WISH)*, Washington DC , 2013.

[5] G. Hammond, T. Croudace, M. Radhakrishnan, L. Lafortune and A. Watson, "Comparative Effectiveness of Cognitive Therapies Delivered Face-to-Face or over the Telephone: An Observational Study Using Propensity Methods," *PLoS ONE,* vol. 7, no. 9, 2012.

[6] D. Mohr, J. Ho, J. Duffecy, D. Reifler, L. Sokol, M. Burns, L. Jin and J. Siddique, "Effect of telephone-administered vs face-to-face cognitive behavioral therapy on adherence to therapy and depression outcomes among primary care patients: a randomized trial.," *JAMA,* vol. 307, no. 21, pp. 2278-85, 2012.

[7] K. Kroenke, R. L. Spitzer and J. B. Williams, "The PHQ-9: validity of a brief depression severity measure.," *J Gen Intern Med,,* vol. 16, no. 9, pp. 606-13, 2001.

[8] J. Ramos-Castro, H. Mahdavi, D. Pérez-Díaz de Cerio, M. A. García-González, S. Estevez and J. Rosell-Ferrer, "Unobtrusive Activity Measurement in Patients with Depression: the Help4Mood Approach," in *Information and Communication Technologies applied to Mental Health, pp.18-23*, Valencia, Spain, 2012.

[9] L. Ferrini and S. Albertini, "Graphical Development of the Help4Mood's Virtual Agent," in *Information and Communication Technologies applied to Mental Health, pp. 40-43*, Valencia, Spain, 2012.

[10] J. Martínez-Miranda, A. Bresó and J. García-Gómez, "The Construction of a Cognitive-Emotional Module for the Help4Mood's Virtual Agent," in *Proceedings of the 1st Workshop on ICT applied to Mental Health*, Valencia. Spain, 2012.

[11] A. Bresó, J. Martínez-Miranda and J. García-Gómez, "A Flexible Computational Architecture for a Personal Health System to Support the Treatment of Depression," *In preparation,* 2013.

[12] C. Sáez, A. Bresó, J. Vicente, M. Robles and J. García-Gómez, "An HL7-CDA wrapper for facilitating semantic interoperability to rule-based Clinical Decision Support Systems," *Computer Methods and Programs in Biomedicine,* vol. 109, no. 3, pp. 239-249, 2013.

[13] M. Wolters, J. Martínez-Miranda, S. Estevez, H. Hastie and C. Matheson, "Managing Data in Help4Mood," *In EAI Endorsed Transactions on Ambient Systems,* vol. 1, pp. 1-6, 2013.

[14] A. X. Garg, N. K. Adhikari, H. McDonald, M. P. Rosas-Arellano, P. J. Devereaux, J. Beyene and R. B. Haynes, "Effects of computerized clinical decision support systems on practitioner performance and patient outcomes," *JAMA: the journal of the American Medical Association,* vol. 293, no. 10, pp. 1223-1238, 2005.

[15] B. Chaudhry, J. Wang and S. Wu, "Systematic review: impact of health information technology on quality, efficiency, and costs of medical care," *Annals of Internal Medicine,* vol. 144, no. 10, pp. 742-52, 2006.

[16] I. Hetlevik, J. Holment, O. Kruger, P. Kristensen, H. Iversen and K. Furuseth, "Implementing clinical guidelines in the treatment of diabetes mellitus in general practice: evaluation of effort, process, and patient outcome related to implementation of a computerbased decision support system.," *International Journal of Technology Assessment in Health Care ,* vol. 16, no. 1, pp. 210-227, 2000.

[17] S. Rahaman and M. S. Hossain, "A belief rule based clinical decision support system to assess suspicion of heart failure from signs, symptoms and risk factors," in *International Conference In Informatics, Electronics & Vision (ICIEV)*, (pp. 1-6). IEEE., 2013.

[18] B. Rollman, B. Hanusa, H. Lowe, T. Gilbert and Kapoor, " A randomised trial using computerized decision support to improve treatment of major depression in primary care.," *Journal of General*

Internal Medicine, vol. 17, pp. 493-503, 2002.

[19] D. Razzouk, J. Mari, I. Shirakawa, J. Wainer and D. Sigulem, "Decision support system for the diagnosis of schizophrenia disorders," *Brazilian Journal of Medical and Biological Research,* vol. 39, no. 1, pp. 119-128, 2006.

[20] M. Trivedi, J. Kern, B. Grannemann, K. Altshuler and P. Sunderajan, "A computerised clinical decision support system as a means of implementing depression guide lines," *Psychiatric Services,* vol. 55, no. 8, pp. 879-885, 2004.

[21] J. C. Fortney, J. M. Pyne, C. A. Steven, J. S. Williams, R. G. Hedrick, A. K. Lunsford and J. L. Smith, "A web-based clinical decision support system for depression care management," *The American Journal of Managed Care,* vol. 16, no. 11, p. 849, 2010.

[22] V. Ekong, U. Inyang and E. Onibere, "Intelligent Decision Support System for Depression Diagnosis Based on Neuro-fuzzy-CBR Hybrid," *Modern Applied Science,* vol. 6, no. 7, 2012.

[23] S. Tortajada, J. García-Gómez, J. Vicente, J. Sanjuán, R. Frutos, R. Martín-Santos, L. García-Esteve, I. Gornemann, A. Gutierrez-Zotes, F. Canellas, A. Carracedo, M. Gratacos, R. Guillamat, E. Baca-García and M. Robles, "Prediction of postpartum depression using multilayer perceptrons and pruning," *Methods of information in medicine,* vol. 48, no. 3, pp. 291-298, 2009.

[24] A. Khodayari-Rostamabad, J. P. Reilly, G. Hasey, H. de Bruin and D. MacCrimmon, "Diagnosis of psychiatric disorders using EEG data and employing a statistical decision model," in *International Conference of Engineering in Medicine and Biology Society (EMBC)*, 2010.

[25] L. Bergman and U. Fors, "BMC medical informatics and decision making," *Decision support in psychiatry - a comparison between the diagnostic outcomes using a computerized decision support system versus manual diagnosis,* vol. 8, no. 9, 2008.

[26] M. Fava, A. Rush, M. Trivedi, A. Nierenberg, M. Thase, H. Sackeim, F. Quitkin, S. Wisniewski, P. Lavori, J. Rosenbaum and D. Kupfer, "Background and rationale for the sequenced treatment alternatives to," *The Psychiatric clinics of North America,* vol. 26, no. 2, pp. 457-94, 2003.

[27] M. L. Crismon, M. Trivedi, T. A. Pigott, A. J. Rush, R. M. Hirschfeld, D. A. Kahn, C. DeBattista, J. C. Nelson, A. A. Nierenberg, H. A. Sackeim and M. Thase, "The Texas Medication Algorithm Project: report of the Texas Consensus Conference Panel on Medication Treatment of Major Depressive," *The Journal of clinical psychiatry,* vol. 60, no. 3, pp. 142-56, 1999.

[28] M. Trivedi and E. Daly, "Measurement-based care for refractory depression: a clinical decision support model for clinical research and practice," *Drug and alcohol dependence,* vol. 88, no. 2, pp. 61-71, 2007.

[29] N. Titov, "Status of computerized cognitive behavioural therapy for adults," *Australian and New Zealand Journal of Psychiatry,* vol. 41, no. 2, pp. 95-114, 2007.

[30] J. Proudfoot, "Beating the Blues," *London: Institute of Psychiatry,* 1999.

[31] D. OSGOOD-HYNES, J. GREIST and I. MARKS, "Self-administered psychotherapy for depression using a telephone-accessed computer system plus booklets: an open US-UK study," *Journal of Clinical Psychiatry,* vol. 59, pp. 358-365, 1998.

[32] D. Haldane, "MoodGYM," *Occupational Medicine,* vol. 56, no. 8, p. 586, 2006.

[33] G. Clarke, E. Reid, D. Eubanks, E. O'Connor, L. DeBar, C. Kelleher, F. Lynch and S. Nunley, "A randomized controlled trial of an internet depression skills intervention program," *Journal of Medical Internet Research,* vol. 4, p. e14, 2002.

[34] E. Kaltenthaler, G. Parry, C. Beverley and M. Ferriter, "Computerised cognitive–behavioural therapy for depression: systematic review," *The British Journal of Psychiatry,* vol. 193, no. 3, pp. 181-184, 2008.

[35] P. S. Foroushani, J. Schneider and N. Assareh, "Meta-review of the effectiveness of computerised CBT in treating depression," *BMC psychiatry,* vol. 11, no. 1, p. 131, 2011.

[36] S. Gerhards, Evaluation of self-help computerised cognitive behavioural therapy for depression; Integrating clinical, economic and patient perspectives, PhD thesis, 2011.

[37] E. Kaltenthaler, J. Brazier, E. De Nigris, I. Tumur, M. Ferriter, C. Beverley, G. Parry, G. Rooney and P. Sutcliffe, "Computerised cognitive behaviour therapy for depression and anxiety update: a systematic review and economic evaluation," *Health Technol Assess,* vol. 10, no. 33, pp. 1-168, 2006.

[38] *PHS2020 Project Personal Health Systems,* FP7-ICT-2007-215291: European Commission, 2007.

[39] A. Bresó, C. Sáez, J. Vicente, F. Larrinaga, M. Robles and J. García-Gómez, "Knowledge-Based Personal Health System to Empower Outpatients of Diabetes Mellitus by means of P4 Medicine," *Expert Systems with Applications,* p. Under Review, 2013.

[40] A. P. Happonen, K. Kaipainen, A. Vtnen, M.-L. Kinnunen, T. Myllymki, P. Lappalainen, H. Tuomela, H. Rusko, E. Mattila, R. Lappalainen and I. Korhonen, "P4Well Concept to Empower Self-Management of

Psychophysiological Wellbeing and Load Recovery," in *4th Int. Conf. Pervasive Comput. Technol. Healthcare*, Munchen Germany, 2010.

[41] D. Krefting, S. Canisiusy, J. Wu, R. Siewert, S. Specovius, K. Kespery and T. Penzel, "Personal Health Systems for Diagnostics of Sleep Disorders using new Sensors and Grid Technology," in *In Digital Ecosystems Technologies (DEST), 6th IEEE International Conference*, pp.1-6, 2012.

[42] R. Giuseppe, R. Baños, C. Botella, A. Gaggioli and B. Wiederhold, "Personal health systems for mental health: the European projects," *Studies in Health Technology and Informatics,* vol. 163, pp. 496-502, 2011.

[43] R. W. Picard, "Toward Machines with Emotional Intelligence.," in *In International Conference on Informatics in Control, Automation and Robotics (ICINCO)*, 29-30, 2004.

[44] A. Baylor, "Promoting motivation with virtual agents and avatars: role of visual presence and appearance," *Philosophical transactions of the Royal Society of London,* vol. 364, no. 1535, pp. 3559-65, 2009.

[45] D. M. Dehn and S. Mulken, "The impact of animated interface agents: a review of empirical research," *International Journal of Human-Computer Studies,* vol. 52, no. 1, pp. 1-22, 2000.

[46] S.-H. Kang, J. Gratch, C. Sidner, R. Artstein, L. Huang and L.-P. Morency, "Towards building a Virtual Counselor: Modeling Nonverbal Behavior during Intimate Self-Disclosure," in *Proceedings of the 11th International Conference on Autonomous Agents and Multiagent Systems*, Valencia, 2012.

[47] T. W. Bickmore, K. Puskar, E. A. Schlenk, L. M. Pfeifer and S. M. Sereika, "Maintaining reality:Relational agents for antipsychotic medication adherence," *Interacting with Computers,* vol. 22, no. 4, pp. 276-288, 2010.

[48] M. Pontier and G. F. Siddiqui, "A virtual therapist that responds empathically to your answers," *Lecture Notes on Artificial Intelligence. 5208(IVA'2008),* pp. 417-425, 2008.

[49] A. Rizzo and B. J. Joseph, "Virtual Reality, A New Therapy Part Two: The SimCoach," *Psychology Today,* 2011.

[50] K. R. Scherer and M. Peper, "Psychological theories of emotion and neuropsychological research," *In Handbook of Neuropsychology,* vol. 5, pp. 17-48, 2001.

[51] K. R. Scherer, "Emotion and emotional competence: conceptual and theoretical issues for modelling agents," in *Blueprint for Affective Computing*, Chapter 1.1, Oxford University Press, 2010, pp. 3-20.

[52] R. S. Lazarus, "Emotion and Adaptation," *NY, Oxford University Press,* 1991.

[53] K. R. Scherer, A. Schorr and T. Johnstone, "Appraisal Processes in Emotion: Theory, Methods, Research," *Oxford University Press,* 2001.

[54] K. R. Scherer, "Emotion and emotional competence: conceptual and theoretical issues for modelling agents," in *Blueprint for Affective Computing: A Sourcebook*, Oxford University Press, 2010, pp. Chapter 1.1, pp. 3-20.

[55] C. Elliot, "The Affective Reasoner: A Process Model of Emotions in a Multi-Agent System.," *PhD thesis,* pp. Institute for the Learning Sciences, Northwestern University, 1992.

[56] J. Dias and A. Paiva, "Feeling and reasoning: A computational model for emotional characters," in *In Proceedings of the 12th Portuguese Conference on Artificial Intelligence (EPIA)*, pages 127–140, 2005.

[57] P. Gebhard, "Alma - a layered model of affect," in *Int. Conf. on Autonomous Agents and MultiAgent Systems (AAMAS)*, 2005.

[58] S. Marsella and J. Gratch, "Ema: A model of emotional dynamics.," *Journal of Cognitive Systems Research,* vol. 10, no. 1, pp. 70-90, 2009.

[59] S. Marsella, J. Gratch and P. Petta, "Computational Models of Emotion," in *Blueprint for afective computing*, chapter 1.2, Oxford University Press, 2010, pp. 21-41.

[60] J. Feblowitz, A. Wright, H. Singh, L. Samal and D. Sittig, "Summarization of clinical information: A conceptual model," *Journal of Biomedical Informatics,* no. 44, p. 688–699, 2011.

[61] E. Fuster-García, J. Juan-Albarracín, A. Bresó and J. Miguel García-Gómez, "Monitoring changes in daily actigraphy patterns of free-living patients," in *International Work-Conference on Bioinformatics and Biomedical Engineering (IWBBIO)*, Granada, Spain, 2013.

[62] G. Moullec, C. Maano, A. Morin, J. Monthuy-Blanc, L. Rosello and G. Ninot, "A very short visual analog form of the Center for Epidemiologic Studies Depression Scale (CES-D) for the idiographic measurement of depression," *Journal of Affective Disorders,* vol. 128, pp. 220-234, 2011.

[63] A. Ortony, G. Clore and A. Collins, "The Cognitive Structure of Emotions," *Cambridge University Press,* 1988.

[64] A. Clark, "Empathy in Counseling and Psychotherapy," in *Perspectives and Practices*, Lawrence

Erlbaum Associates, 2007.

[65] S. Meier and S. Davis, The elements of counseling (5th ed.)., Belmont: Brooks/Cole, 2004.

[66] J. Gross, HandBook of emotion regulation, New York: Guilford Press, 2007.

[67] A. Axmisan, A. McCoy, A. Wright and D. Sittig, "Clinical Summarization Capabilities of Commercially-available and Internally-developed Electronic Health Records," *Applied Clinical Informatics,* vol. 3, pp. 80-93, 2012.

[68] R. Thwaites and J. Bennett-Levy, "Conceptualizing empathy in cognitive behaviour therapy: Making the implicit explicit," *Behavioural and Cognitive Psychotherapy,* vol. 35, pp. 291-612, 2007.

[69] I. Chartier and M. Provencher, "Behavioural activation for depression: efficacy, effectiveness and dissemination," *Journal of affective disorders,* vol. 145, no. 3, pp. 292-9, 2013.

[70] I. Gotlib, E. Krasnoperova, D. Yue and J. Joormann, "Attentional biases for negative interpersonal stimuli in clinical depression.," *Journal of abnormal psychology,* vol. 113, no. 1, pp. 121-35, 2004.

[71] S. Baert, R. Raedt, R. Schacht and E. Koster, "Attentional bias training in depression: therapeutic effects depend on depression severity," *Journal of behavior therapy and experimental psychiatry,* vol. 41, no. 3, pp. 265-74, 2010.

Annex A - Available defined tasks in the system

Code	Task Description	Task Name
30000011	The system greets the patient if the last login occurs within the 3 days (the value of the days can be included as parameter in the task structure)	**Welcome**
30000012	Welcome and reminder to the user that is recommended to use the system regularly. It's inferred if the last login occurred in more than 3days.	**Welcome_Reminder**
30000031	Question number 1 of the Daily Mood Check (CES-VAS-VA): "How is your mood today?"	**DailyMoodCheck_1**
30000032	Question number 1 of the Daily Mood Check (CES-VAS-VA): "I have felt happy"	**DailyMoodCheck_2**
30000034	Question number 1 of the Daily Mood Check (CES-VAS-VA): "I have talked less than usual"	**DailyMoodCheck_3**
30000036	Question number 1 of the Daily Mood Check (CES-VAS-VA): "I have felt like crying"	**DailyMoodCheck_4**
30000038	Question number 1 of the Daily Mood Check (CES-VAS-VA): "I feel that people dislike me"	**DailyMoodCheck_5**
30000081	Activity monitoring dialog it's inferred if the DSS detect that the PMS is not reporting sensor data in the last 3 days (the value of the days can be included as parameter in the task structure) in order to recommend to use the sensor devices.	**Activity_Monitoring**
30000121	Normal goodbye session.	**Farewell**
30000131	It indicates the end of the execution of the system. It's the last tasks of the daily session.	**Shutdown**
30000140	Ask if the patient wants to start the NTQ task.	**Negative_Thoughts_Activity _Introduction**
30000141	Ask if there has been a negative situation	**Elicit_Negative_Situation**
30000142	Ask for negative situation domain: (1) How you felt, your wellbeing (2) Relationships & events at home (3) Event & relationships outside the home	**Request_Negative_Situation_Do main**
30000143	Ask for negative situation inference: (1) What I thought about myself (2) What other people think about me (3) How I think about the future	**Request_Negative_Situation_Inf erence**
30000144	Ask if it is possible to describe the negative situation	**Ask_For_Negative_Situation_Te xt**
30000145	Request the free text describing the negative situation	**Request_Negative_Situation_Te xt**
30000146	It shows to the patient some negative thoughts. The patient must select one of them.	**Select_Negative_Thought**
30000147	The patient needs more negative thoughts choices	**Select_Negative_Thought_Again**
30000148	It shows an alternative thought from the negative thought.	**Provide_Alternative_Thought**
30000149	It confirms the selection	**Accept_Alternative_Thought**
30000150	Ask if it is possible to provide some alternative thoughts	**Ask_Provide_Alternative_Thoug ht**
30000151	Ask if there has been a positive situation	**Elicit_Positive_Situation**
30000152	Ask for positive situation domain: (1) How you felt, your wellbeing (2) Relationships & events at home (3) Event & relationships outside the home	**Request_Positive_Situation_Do main**
30000153	Ask for positive situation inference: (1) It helped what I thought about myself (2) It showed what other people think about me (3) It showed	**Request_Positive_Situation_Infe rence**

	something positive about the future	
30000154	Ask if it is possible to describe the positive situation	**Ask_For_Positive_Situation_Text**
30000155	Request the free text describing the positive situation	**Request_Positive_Situation_Text**
30000156	Ask if it is possible to describe the positive situation	**Ask_Explore_Positive_Situation**
30000157	It shows to the patient some positive thoughts. The patient must select one of them.	**Select_Positive_Thought**
30000158	The patient needs more positive thoughts choices	**Select_Positive_Thought_Again**
30000191	End of "Negative Thoughts" activity	**End_Negative_Thoughts_Activity**
30000200	Task to ask the user whether to start the PHQ9 questionnaire (frequency: each 7 days)	**Introduce_PHQ9**
30000201	It shows the PHQ9 task to the user.	**Request_PHQ9**
30000300	This task proposes to the patient three issues in order to record the patient voice.	**Speech_Activity**
30000091	This task is launched when the total score of the DMC questionnaire or the question number 9 of the PHQ9 is under the threshold.	**Crisis_Plan**
30000092	This task is launched when the user press crisis plan button	**Crisis_Plan_User_Button**
10000040	This task proposes to the user the length of the session: Long, Medium, or Short. (related with patient stamina)	**Select_Session_Type**
10000041	This sleep questionnaire contains 9 questions (qualitative and quantitative)	**Sleep_Questionnaire**
30000093	User press logout button	**Logout_User_Button**
30000094	User press Send_Report button	**Send_Report_User_Button**
10000101	"This is the report…"	**Display_Report**
10000100	"The report is ready, do you want view it?"	**Report_Ready**
10000102	The report is generated and store in the hard disk	**Generate_Report**

Table 8: Available defined tasks in the system

Annex B - UserInputFromDM.xsd template (data coming from GUI to DASPS)

```xml
<?xml version="1.0" encoding="ISO-8859-1" ?>
<!-- UserInputFromDM -->
<xs:schema xmlns="urn:userInputFromDM.data.ke.dss.h4m.bmg.ibime.upv.es"
targetNamespace="urn:userInputFromDM.data.ke.dss.h4m.bmg.ibime.upv.es"
xmlns:xs="http://www.w3.org/2001/XMLSchema"
xmlns:com="urn:commonTypes.data.dss.h4m.bmg.ibime.upv.es" xmlns:jaxb="http://java.sun.com/xml/ns/jaxb"
jaxb:version="2.0">
<xs:annotation>
<xs:appinfo>
<jaxb:globalBindings>
<jaxb:javaType name="java.util.Calendar" xmlType="xs:dateTime"
parseMethod="javax.xml.bind.DatatypeConverter.parseDateTime"
printMethod="javax.xml.bind.DatatypeConverter.printDateTime" />
</jaxb:globalBindings>
</xs:appinfo>
</xs:annotation>
<xs:import namespace="urn:commonTypes.data.dss.h4m.bmg.ibime.upv.es"
schemaLocation="commonTypes.xsd"/>
<!--userInputFromDM-->
<xs:element name="userInputFromDM">
<xs:complexType>
<xs:sequence>
<!--Description-->
        <xs:element name="description" type="com:Description" minOccurs="1" maxOccurs="1"/>
<!--timestamp-->
        <xs:element name="timestamp" type="xs:dateTime" minOccurs="1" maxOccurs="1"/>
<!--result-->
        <xs:element name="result" minOccurs="1" maxOccurs="1">
<xs:simpleType>
<xs:restriction base="xs:string">
<xs:enumeration value="OK"/>
<xs:enumeration value="NO"/>
</xs:restriction>
</xs:simpleType>
</xs:element>
<xs:element name="input" minOccurs="0" maxOccurs="1" type="Input"/>
</xs:sequence>
</xs:complexType>
</xs:element>
<!-- Input -->
<xs:complexType abstract="true" name="Input"/>
<!--Questionnaire-->
        <xs:complexType name="Questionnaire">
        <xs:complexContent>
        <xs:extension base="Input">
        <xs:sequence>
        <!--scores-->
        <xs:element name="scores">
        <xs:complexType>
        <xs:sequence>
        <!--score-->
        <xs:element name="score" type="Score" minOccurs="0"    maxOccurs="unbounded"/>
        </xs:sequence>
        </xs:complexType>
        </xs:element>
        </xs:sequence>
        </xs:extension>
        </xs:complexContent>
        </xs:complexType>
        <xs:complexType name="Score">
        <xs:complexContent>
```

```xml
            <xs:extension base="Input">
            <xs:sequence>
            <!--questionNumber-->
            <xs:element name="questionNumber" type="xs:int"/>
            <!--value-->
            <xs:element name="value" type="xs:string"/>
            </xs:sequence>
            </xs:extension>
            </xs:complexContent>
            </xs:complexType>
<!--Diary-->
            <xs:complexType name="Diary">
            <xs:complexContent>
            <xs:extension base="Input">
            <xs:sequence>
            <!--length-->
            <xs:element name="length" type="xs:int"/>
            </xs:sequence>
            </xs:extension>
            </xs:complexContent>
            </xs:complexType>
<!--Speech-->
            <xs:complexType name="Speech">
            <xs:complexContent>
            <xs:extension base="Input">
            <xs:sequence>
            <!--changeInSpeaking
            <xs:element name="changeInSpeaking">
            <xs:simpleType>
            <xs:restriction base="xs:int">
            <xs:enumeration value="0"/>
            <xs:enumeration value="1"/>
            <xs:enumeration value="-1"/>
            </xs:restriction>
            </xs:simpleType>
            </xs:element>
            <!--changeInIntonation-->
            <xs:element name="changeInIntonation">
            <xs:simpleType>
            <xs:restriction base="xs:int">
            <xs:enumeration value="0"/>
            <xs:enumeration value="1"/>
            <xs:enumeration value="-1"/>
            </xs:restriction>
            </xs:simpleType>
            </xs:element>
            <!--WavFilePath-->
            <xs:element name="WavFilePath" type="xs:string"/>
            </xs:sequence>
            </xs:extension>
            </xs:complexContent>
            </xs:complexType>
<!--Game-->
            <xs:complexType name="Game">
            <xs:complexContent>
            <xs:extension base="Input">
            <xs:sequence>
            <!--changeInReaction-->
            <xs:element name="changeInReaction">
            <xs:simpleType>
            <xs:restriction base="xs:int">
            <xs:enumeration value="0"/>
            <xs:enumeration value="1"/>
            <xs:enumeration value="-1"/>
            </xs:restriction>
            </xs:simpleType>
            </xs:element>
            <!--changeInScore-->
            <xs:element name="changeInScore">
            <xs:simpleType>
            <xs:restriction base="xs:int">
            <xs:enumeration value="0"/>
```

```
                <xs:enumeration value="1"/>
                <xs:enumeration value="-1"/>
            </xs:restriction>
        </xs:simpleType>
    </xs:element>
    </xs:sequence>
    </xs:extension>
    </xs:complexContent>
    </xs:complexType>
<!--Selector-->
        <xs:complexType name="Selector">
        <xs:complexContent>
        <xs:extension base="Input">
        <xs:sequence>
        <!--selections-->
        <xs:element name="selections">
        <xs:complexType>
        <xs:sequence>
        <!--selection-->
        <xs:element name="description" type="com:Description" minOccurs="1" maxOccurs="unbounded"/>
        </xs:sequence>
        </xs:complexType>
        </xs:element>
        </xs:sequence>
        </xs:extension>
        </xs:complexContent>
        </xs:complexType>
<!--Free Text-->
        <xs:complexType name="FreeText">
        <xs:complexContent>
        <xs:extension base="Input">
        <xs:sequence>
        <!--inputText-->
        <xs:element name="inputText" type="xs:string"/>
        </xs:sequence>
        </xs:extension>
        </xs:complexContent>
        </xs:complexType>
</xs:schema>
```

67

Annex C - TaskPlanning.xsd template

```xml
<?xml version="1.0" encoding="UTF-8"?>
<!-- tasksPlanning SCHEMA -->
<xs:schema xmlns="urn:tasksplanning.data.sp.dss.h4m.bmg.ibime.upv.es"
targetNamespace="urn:tasksplanning.data.sp.dss.h4m.bmg.ibime.upv.es"
xmlns:xs="http://www.w3.org/2001/XMLSchema"
xmlns:com="urn:commonTypes.data.dss.h4m.bmg.ibime.upv.es"
xmlns:jaxb="http://java.sun.com/xml/ns/jaxb" jaxb:version="2.0">

<xs:annotation>
<xs:appinfo>
<jaxb:globalBindings>
<jaxb:javaType name="java.util.Calendar" xmlType="xs:dateTime"
parseMethod="javax.xml.bind.DatatypeConverter.parseDateTime"
printMethod="javax.xml.bind.DatatypeConverter.printDateTime" />
</jaxb:globalBindings>
</xs:appinfo>
</xs:annotation>
<xs:import namespace="urn:commonTypes.data.dss.h4m.bmg.ibime.upv.es" schemaLocation="commonTypes.xsd"/>

<!--tasksPlanning-->
<xs:element name="tasksPlanning">
<xs:complexType>
<xs:sequence>
<!--item-->
<xs:element name="item" type="Item" minOccurs="0" maxOccurs="unbounded"/>
</xs:sequence>
</xs:complexType>
</xs:element>

<!-- each item defines one candidateActivity which will be planned by the Session Planer -->
<!--Item  -->
<xs:complexType name="Item">
        <xs:sequence>
        <!--description-->
        <xs:element name="description" type="com:Description" minOccurs="1" maxOccurs="1"/> <!-- code,
snomed, name -->
        <!--enable-->
        <xs:element name="enable" minOccurs="1" maxOccurs="1">
        <xs:simpleType>
                <xs:restriction base="xs:string">
                <xs:enumeration value="YES"/>
                <xs:enumeration value="NO"/>
        </xs:restriction>
        </xs:simpleType>
        </xs:element>
        <!--typeOfItem-->
        <xs:element name="typeOfItem" minOccurs="1" maxOccurs="1">
        <xs:simpleType>
                <xs:restriction base="xs:string">
                <xs:enumeration value="REQUIRED"/>
                <!-- These activities don't need the MinFrec/MaxFrec because they will be plan allways -->
                <xs:enumeration value="OPTIONAL"/>
                <!-- These need the MinFrec/MaxFrec in order to decide if they will be plan or not -->
        </xs:restriction>
        </xs:simpleType>
        </xs:element>
        <!--priority-->
        <xs:element name="priority" type="xs:int" minOccurs="1" maxOccurs="1"/> <!-- 1...100 -->
        <!--minFrec-->
        <xs:element name="minFrec" type="xs:int" minOccurs="0" maxOccurs="1"/> <!-- 1...7 -->
        <!--maxFrec-->
        <xs:element name="maxFrec" type="xs:int" minOccurs="0" maxOccurs="1"/> <!-- 1...7 -->
        <!--codeOfDataInDB-->
        <xs:element name="codeOfDataInDB" type="xs:string" minOccurs="0" maxOccurs="1"/>
```

69

```xml
        <!-- These atribute is needed if item is Optional.The system use this atribute in order to looking for the past
executionons of the activity. With this information, the SP is enable to plan or not the item depending of the
MinFrec/MaxFrec.-->
        <!--repeatable-->
        <xs:element name="repeatable" minOccurs="1" maxOccurs="1">
        <xs:simpleType>
                <xs:restriction base="xs:string">
                <xs:enumeration value="YES"/>
                <xs:enumeration value="NO"/>
        </xs:restriction>
        </xs:simpleType>
        </xs:element>
        <!--constrains-->
        <xs:element name="constrains">
        <xs:complexType>
        <xs:sequence>
        <!--constrain-->
        <xs:element name="constrain" type="Constrain" minOccurs="0" maxOccurs="unbounded"/>
        </xs:sequence>
        </xs:complexType>
        </xs:element>
        </xs:sequence>
        </xs:complexType>
        <!--Constrain-->
        <xs:complexType name="Constrain">
        <xs:sequence>
        <!--constrainDescriptions-->
        <xs:element name="constrainDescription" type="com:Description" minOccurs="1" maxOccurs="1"/> <!--
code, snomed, name -->
        <!--constrainType-->
        <xs:element name="constrainType" minOccurs="1" maxOccurs="1">
        <xs:simpleType>
        <xs:restriction base="xs:string">
        <xs:enumeration value="AFTER"/> <!--If this exists, it must plan AFTER than original activity-->
        <xs:enumeration value="BEFORE"/> <!--If this exists, it must plan BRFORE than original activity-->
        <xs:enumeration value="BLOCKBY"/> <!--If this exists, the original activity is not planning-->
        <xs:enumeration value="ENABLEBY"/> <!--This must exists, in order to plan the original activity-->
        </xs:restriction>
        </xs:simpleType>
        </xs:element>
</xs:sequence>
</xs:complexType>
</xs:schema>
```