FACULDADE DE ENGENHARIA DA UNIVERSIDADE DO PORTO

Personal Medication Advisor

Joana Polónia Lobo

MASTER IN BIOENGINEERING

Supervisor at FhP: Liliana Ferrreira (PhD)
Supervisor at FEUP: Aníbal Ferreira (PhD)

July, 2013

Personal Medication Advisor

Joana Polónia Lobo

Master in Bioengineering

Approved in oral examination by the committee:

Chair: Artur Cardoso (PhD)

External Examiner: Fernando Perdigão (PhD)

Supervisor at FhP: Liliana Ferrreira (PhD)

Supervisor at FEUP: Aníbal Ferreira (PhD)

July, 2013

Abstract

Healthcare faces new challenges with automated medical support systems and artificial intelligence applications for personal care. The incidence of chronic diseases is increasing and monitoring patients in a home environment is inevitable. Heart failure is a chronic syndrome and a leading cause of death and hospital readmission on developed countries. As it has no cure, patients will have to follow strict medication plans for the rest of their lives. Non-compliance with prescribed medication regimens is a major concern, especially among older people. Thus, conversational systems will be very helpful in managing medication and increasing adherence.

The Personal Medication Advisor aimed to be a conversational assistant, capable of interacting with the user through spoken natural language to help him manage information about his prescribed medicines. This patient centred approach to personal healthcare will improve treatment quality and efficacy. System architecture encompassed the development of three modules: a language parser, a dialog manager and a language generator, integrated with already existing tools for speech recognition and synthesis. All these modules work together and interact with the user through an Android application.

System evaluation was performed through a usability test to assess feasibility, coherence and naturalness of the Personal Medication Advisor. It was concluded that the developed assistant could successfully make users achieve their objectives. In addition, the system was evaluated as easy to use, coherent and with a pleasant interface. Further improvements can still be done to extend the flexibility and convenience of the Personal Medication Advisor so that an easy integration on senior patient’s daily life can be achieved.

Resumo

Os cuidados de Saúde estão a ser revolucionados pela introdução de sistemas automáticos de suporte à decisão médica e outras aplicações de inteligência artificial. A incidência de doenças crónicas está a aumentar e torna-se inevitável monitorizar os pacientes em casa. A insuficiência cardíaca é uma síndrome crónica, sendo a principal causa de morte e readmissão hospitalar nos países desenvolvidos. Como não tem cura, os pacientes terão de seguir exigentes regimes de medição para o resto das suas vidas. O não cumprimento dos planos de medicação receitados é também uma preocupação, especialmente entre pacientes mais idosos. Assim, assistentes conversacionais poder-se-ão tornar muito úteis para gerir a medicação e aumentar a adesão aos planos prescritos.

O Assistente Pessoal de Medicação procura ser um assistente conversacional capaz de interagir através de linguagem natural falada com o utilizador para o ajudar a gerir os seus medicamentos. Esta visão centrada no paciente pretende aumentar a eficácia e qualidade dos cuidados de Saúde. A arquitetura do sistema passou pelo desenvolvimento de três módulos: um processador de linguagem, um gestor de diálogo e um gerador de linguagem, integrados com ferramentas já existentes para reconhecimento e síntese de voz. Todos estes módulos atuam em conjunto e interagem com o utilizador através duma aplicação para Android.

A avaliação do sistema recorreu a um teste de usabilidade para qualificar a performance, coerência e naturalidade do Assistente Pessoal de Medicação. O assistente desenvolvido conseguiu fazer com que os utilizadores atingissem os objetivos pretendidos. A aplicação foi também classificada como fácil de usar, coerente e com uma interface agradável. Futuras melhorias no sistema devem ser realizadas para aumentar a flexibilidade e conveniência do Assistente Pessoal de Medicação. Deste modo, será conseguida uma melhor e mais facilitada integração do assistente no dia-a-dia de pacientes idosos.

Acknowledgments

My thanks to Dr. Lilliana Ferreira, supervisor at Fraunhofer Portugal, and Prof. Aníbal Ferreira, supervisor at FEUP, for providing the required support and guidance during this project.

To Ana Barros for her good advices and to Bruno Aguiar for letting me use his *PharmInx* ontology.

To Fraunhofer Portugal AICOS for the opportunity of developing this project and to those who volunteered for testing the Personal Medication Advisor or simply gave their suggestions for the project.

A special thanks to all my friends and family.

To my mom and boyfriend who support me daily: “Gratitude is the memory of the heart.”

Joana Polónia Lobo

“*The best way to predict the future is to invent it.*”

Alan Kay

Contents

List of Figures

List of Tables

Abbreviations

Abbreviation list (by alphabetic order)

ACE	Angiotensin-converting enzyme
ADT	Android development tools
ARB	Angiotensin receptor blocker
ASR	Automated speech recognition
API	Application programmatic interface
CHF	Chronic heart failure
EF	Ejection fraction
ESC	European society of Cardiology
HF	Heart failure
HF-PEF	Heart failure with preserved ejection fraction
HF-REF	Heart failure with reduced ejection fraction
H-ISDN	Hydralazine and isosorbide dinitrate
JSGF	Java speech grammar format
LKB	Linguistic knowledge database
MEMS	Medication event monitoring system
MRA	Mineralocorticoid receptor antagonist
MTT	Meaning text theory
NLG	Natural language generation
NLU	Natural language understanding
OWL	Web ontology language
PMA	Personal medication advisor
PUFA	3-n polyunsaturated fatty acid
RDF	Resource description framework
SPARQL	SPARQL protocol and RDF query language
SUS	System usability scale
TTS	Text-to-speech
VPL	Voice prescription labels

Chapter 1

Introduction

Health care technologies seem to be evolving towards interactive and personal health management assistants. Interaction with conversational assistants should be seen as an ongoing dialogue between user and system. Therefore, user's intentions and goals must be understood by the system so that proactive help can be provided. Such technologies require a high level of system architecture as well as a complex coordination between its different components. This work intended to take another step on the direction of a Personal Medication Advisor (PMA), a conversational system capable of interacting with the user through natural language, mostly voice, to help him managing his prescribed medication regimens. This development was placed under the scope of a Dissertation project for the Integrated Master in Bioengineering at Faculty of Engineering of University of Porto and in association with Fraunhofer Portugal Research Center for Assistive Information and Communication Solutions (AICOS).

This chapter firstly introduces the context, scope and needs behind conversational assistants for healthcare. Then, the motivation and objectives that supported the Personal Medication Advisor project are presented, as well as the requirements envisioned for the system and the anticipated benefits and innovative contributions of this research. Finally, the structure of this thesis is outlined.

1.1 The problem

As modern Medicine is developing, a large amount of pharmacological drugs continues to appear. The amount of money annually spent on medicines is growing exponentially (Sullivan, Behncke, & Purushotham, 2010). With the existence of so many drugs and possible medication regimens, Pharmacology brings a large amount of information to be managed by healthcare staff and patients.

While developed world population is aging, there are more demanding health care requirements so that elders can have quality of life. As age increases, the number of patients suffering from chronic illnesses is also raising (Giles, 2004). Heart failure (HF) is a common chronic disease with high prevalence and significant health expenditure due to frequent patient re-hospitalizations (Michalsen, König, & Thimme, 1998). Such diseases usually have numerous pharmacological prescriptions that require strict medication management. With complex medication regimens, patients spend a lot of effort gathering the information required to successfully manage their disease. Most of them even have trouble keeping track of this information. (Ferguson, et al., 2010) Due to unaffordable costs and insufficient medical personnel, healthcare staff cannot offer daily home-assistance. Therefore, there is a need of automated systems for health communication so that intelligence assistance can be provided to patients within a home health environment.

Compliance is an important predictor of outcome among patients with chronic diseases like HF. As slight improvements in functional capacity are of the utmost importance to these patients, prescribed treatments should be taken seriously. Studies have shown that, in general, medication adherence is remarkably lower than self-reported adherence (Nieuwenhuis, Jaarsma, van Veldhuisen, & van der Wal, 2012). With this, it is also essential to maintain patient motivation and promote adherence to the pharmacological regimens in chronic patients.

1.2 Motivation and objectives

On the last decade, healthcare has arrived to the next stage of evolution with automated medical support systems and personal care applications (Topol, 2010) (Haux, 2006). The role of this work was to encompass this next technological step with a Personal Medication Advisor in harmony with Fraunhofer Portugal AICOS mission of Information and Communication Technologies for ‘care, well-being and inclusion’.

Numerous pharmacological treatments rely on strict medication programs that should be complemented with lifestyle measures like dietary changes and appropriate exercise protocols. Therefore, conversational medical assistance may be very helpful, especially with chronic patients and elder healthcare. This work aimed to provide information and knowledge-based advice to help the user managing complex prescription regimens. Personalized intelligence assistance on posology, interactions, indications and adverse reactions will allow in-home medication monitoring. Since many patients do not recognize the value of strictly following medication regimens (Nieuwenhuis, Jaarsma, van Veldhuisen, & van der Wal, 2012), another objective includes motivating the user through interaction and dialogue, trying to increase medication adherence.

The scope of the conversational assistant developed was restricted to medication regimens for heart failure. This syndrome was chosen due to its well defined pharmacological practices which greatly depend on self-care and self-management to lengthen patient survival. Since there is no cure for HF, patients will likely have to follow a medication plan for the rest of their lives.

Although expensive, existing HF home-monitoring programs have proved to be very helpful (Simon & Horowitz, 2002). A personal health assistant to interactively manage medication and prescriptions would help expanding monitoring access for HF patients by reducing healthcare costs, re-hospitalizations and number of face-to-face consultations.

Dialogue systems for health communication generally aim to provide intelligence assistance to patients through spoken natural language conversation within a home health environment. Looking for this final achievement, the main goal of this project was to develop a modular system that would understand what the patient says and produce valid sentences, on a context of heart failure medication regimens, answering the user concerns. The integration of language understanding and generator modules with existing voice recognition and synthesis systems allows spoken interaction with the user. To complement this communication, information through images or text can also be available, but never as the main focus of the system-user interaction.

1.3 Requirements

As an agent-based spoken dialogue system, the Personal Medication Advisor mission tried to accomplish a 'Remarkable Technology, Easy to Use' motto. Under this thought, the PMA was built on the paradigms of consistency and simplicity where the generated sentences are intended to be short, direct and easy to understand. The goal was to make the system reliable, intuitive and portable. The system should not be too constraining, being at the same time easy to use. Because a decisive factor for adoption of technology is convenience (Jimison, et al., 2008), the system tried to require no special training for its utilization. The ultimate delivery to the consumers should be made using an everyday platform like the Android technology.

Real-time generation is essential on a dialogue framework. With this requirement in mind, all the system modules had to be reasonably fast so that the PMA can be used on a real healthcare environment. Moreover, the generation had to be psycho-linguistically realistic and stylistically appropriated. Generated phrases must give useful information and advice on prescribed medication posology for heart failure scenarios. The outputted speech must be well-formed, using formal but short and direct sentences.

Taking these requirements in consideration, the medication assistant is settled to provide information on:

- Posology - Information on drug administration methods and quantities. Subcategories of posology data include administration dosage, route, frequency, drug name and time references. This information on the drug-posology concept was classified as the most important and it should be incorporated during the first step of system development.
- Drug indications - Specification of common targets and uses for a certain pharmacological drug.

- Adverse reactions - Information on the side-effects and harms of the usage of a pharmacological drug under the recommended administration conditions.

A later and advanced approach to the Personal Medication Assistant should also manage contraindications, cautions and drug to drug interactions.

The Personal Medication Advisor was developed in European Portuguese language, opening an unexplored path on the integration of intelligence assistance in Portuguese healthcare.

1.4 Benefits

The Personal Medication Advisor partially shares its beliefs with 'Information therapy'. This methodology defends that providing specific evidence-based medical information to certain patients at the right time can help them making positive behaviour changes and improve self-consciousness of healthcare (Mettler & Kemper, 2003). The patient centred approach to health services seems to be a key to improve disease outcomes. Tailored health information and personalized health education are thus essential for effective healthcare (Jones, Cawsey, & Bentala, 1999). It is expected that the conversational assistant developed can bring together the advantages of information therapies and automated technologies, improving the effectiveness of health services, especially for people with chronic diseases or advanced age. Since chronic diseases, like HF, require strict and complex medications regimens, they are usually a potential menace to patient life and it was suggested that using automated intelligence assistance to manage medication could help in patient survival (Ferguson, et al., 2010). Therefore, it is anticipated that efficiency of healthcare services is enhanced by increasing patient survival, reducing re-hospitalization and mortality rates, diminishing complications and allowing a better responsiveness and distribution of work time for healthcare staff. Patients will also be able to live more independently with greater compliance and satisfaction with therapeutic regimens. The contribution of new technological solutions will surely lead to a pro-active attitude towards disease monitoring and management.

1.5 Outline

After presenting the aim, context and intents of the Personal Medication Advisor project, this thesis continues with *Chapter 2*, containing some background on heart failure, healthcare monitoring and conversational systems. The information presented is needed so that project context and design can be fully understood. The subsequent chapter, *Chapter 3*, identifies, characterizes and justifies the resources and tools used, opening the path to *Chapter 4*, where the PMA architecture, along with each module and feature, is explained in detail. Consequently, the evaluation performed and a discussion around the final prototype of PMA can be seen in *Chapter 5*, with special focus on the difficulties and challenges encountered. Finally, *Chapter 6* presents a final balance of the five months of project development, concluding on what was achieved and what could still be done in the future.

Chapter 2

Background and Literature Review

In this chapter, a brief review of Heart Failure, healthcare technologies and conversational systems is presented. The provided background intends to face up to the aims and motivations of the Personal Medication Advisor, so that the final solution could be based on a systematic literature review.

2.1 Heart Failure

2.1.1. The Disease

The European Society of Cardiology (ESC) defines Heart Failure (HF) as an "abnormality of cardiac structure or function leading to failure of the heart to deliver oxygen at a rate that fulfils the requirements of metabolizing tissues". This syndrome is characterized by typical symptoms as fatigue, coughing, breathlessness or ankle swelling, and signs like pulmonary crackles, a third heart sound or elevated jugular venous pressure. There are several diagnostic methodologies available from essential tests as chest X-ray, electro- and echo-cardiogram to blood analysis, exercise testing and other imaging techniques (cardiac magnetic resonance, positron emission spectroscopy, cardiac computed tomography or coronary angiography). Even so, HF is often difficult to diagnosis, especially in early stages. Many of the symptoms are non-specific and, thus, with limited diagnostic value. At the same time, many of the signs can result from congestion (i.e. sodium and water retention) being absent in patients following diuretic therapies. Precise identification of the cardiac problem is then crucial to determine the specific pharmacological therapy. Symptoms and signs are also essential to secure and monitor the response and stability of HF patients over time (European Society of Cardiology, 2012).

Heart Failure can be divided in systolic (HF-REF) or preserved (HF-PEF) according to the underlying cardiac abnormality and the left ventricular ejection fraction (EF). Other classifications of HF as acute or chronic can also be done. However, a detailed description of all types of HF does not serve the purpose of this work. As HF-REF is the best understood and most common type of HF (European Society of Cardiology, 2012), it will be the focus of this work.

2.1.2. Aetiology & Epidemiology

There are many possible triggers of HF, from diseases with potential to cause cardiac damage to certain nutritional behaviours or unhealthy lifestyles (*Appendix A.I*). HF is unusual in individuals with irrelevant medical history. Causes of this syndrome can also vary according to the world region and social environment (European Society of Cardiology, 2012).

Similar statistics in Europe and USA associate primary diagnosis of chronic HF (CHF) with one million hospitalizations every year in each region (Heart Failure Association of the ESC, 2010). In Portugal, general prevalence of CHF in adults over 25 years old was estimated to be 4.36% in 1999 (Ceia, et al., 2004). Overall, 1-2% of the adult population in developed countries suffers from HF (European Society of Cardiology, 2012). This disease was identified as the leading reason for hospitalization in people older than 65 years. (Krumholz, et al., 2000) This age-related prevalence of HF is rising to more than 10% among persons older than 70 years of age (European Society of Cardiology, 2012).

2.1.3. Impact

HF reality is described by high mortality, morbidity and a recently large increase in HF-related healthcare costs in Europe (Heart Failure Association of the ESC, 2010). A study for UK showed that, in 2000, total HF-related healthcare expenditure were approximately £905 million (1.078€ million), almost 2% of total National Health Service expenditure. Hospital stays and readmissions represented the major cost-component of expenditure related to the overall management of heart failure (Stewarta, et al., 2001).

The development of HF symptoms and their worsening over time will unquestionably mean reduced quality of life for the patient with progressive declining of functional capacity. Frequent episodes of decompensation lead to recurrent hospital admissions and can even culminate in premature death. Before the existence of effective treatments, 60-70% of HF patients were often readmitted to hospital with worsening symptoms. With new pharmacological treatments appearing in the 90s, re-hospitalization was reduced in 30%-50% and mortality slightly decreased (European Society of Cardiology, 2012).

2.1.4. Pharmacological Treatments

The ESC provides updated guidelines for diagnosis and treatment of HF-REF (European Society of Cardiology, 2012). Treatments aim to prevent or slow gradual worsening of the syndrome and assure patient survival. The ESC established three main goals for HF treatments: relieve of symptoms and signs improving quality of life, prevent hospital admissions and improve patient survival. Preventing recurrent hospitalization is important not only for the patients but also for healthcare systems. There are already effective pharmacological therapies to improve disease outcomes although they are not always successful.

A treatment strategy for the use of drugs in patients with HF-REF is well-established and recommended worldwide. Three neurohumoral antagonists (an angiotensin-converting enzyme inhibitor or an angiotensin receptor blocker, a beta-blocker and a mineralocorticoid receptor antagonist MRA) are important to change the course of the disease and must be mandatory in every HF patient. Simultaneously a diuretic is used to relieve the symptoms and signs of congestion. Evidence states that these substances should be given to the patient as soon as possible after diagnosis (European Society of Cardiology, 2012). Reductions in mortality and hospitalization rates reflect the efficiency of treatments.

- *Angiotensin-converting enzyme (ACE) inhibitors*

This group of drugs causes dilation of blood vessels lowering blood pressure, reducing ventricular hypertrophy. They improve symptoms, quality of life and exercise tolerance and performance contributing also to reduce mortality rates. Occasionally, these drugs can worsen renal function and cause symptomatic hypotension, cough, hyperkalaemia and, rarely, angioedema. Due to these reasons, they are not used in patients with inadequate renal function and without a normal serum potassium level. If it is extremely necessary to give the drug to these patients, it is mandatory to consistently monitor renal function and serum electrolytes. Special attention is needed with the elderly. The most used ACE inhibitors are Enalapril, Ramipril, Captopril, Lisinopril and Trandolapril.

- *Angiotensin receptor blockers (ARBs)*

There are no longer many doctors that prescribe ARBs instead of ACE inhibitors. Currently, these drugs are used to substitute ACE inhibitors if patients are unable to tolerate them, especially due to cough or other of the already mentioned adverse reactions. If a MRA is not tolerable, ARB treatments can also be recommended. Candesartan, Valsartan and Losartan are some of the ARBs often used.

- *Beta-blockers*

These substances often have a substantial effect in improving the EF. They also have an anti-ischaemic effect, reducing the risk of sudden cardiac death. This is the main contributor to the reduction in mortality within one year of treatment. It should be used with caution in patients after or during decompensation episodes where dose reduction may be necessary. The most common beta-blockers are Bisoprolol, Carvediol, Nebivolol and Metoprolol succinate.

- *Mineralocorticoid receptor antagonists (MRAs)*

MRAs inhibit sodium reabsorption in the kidneys interfering with sodium and potassium exchanges. The main MRAs are Spironolactone and Eplerenone, block receptors that bind aldestorone and other corticosteroids. It was shown that they have reduced mortality and hospitalization rates for patients after 2 years of treatment. Like the ACE inhibitors, they can cause hyperkalaemia and worsening of the renal function. Spironolactone can also cause breast discomfort and enlargement in men.

- *Diuretics*

They do not directly reduce HF mortality or hospitalization rates. These substances are only used to relieve congestion symptoms as well as achieve and maintain the 'dry weight' of the patient. The strength of the diuresis will depend on the dose, type of diuretic used and the patient renal function. Drug doses should be well-adjusted to avoid dehydration, hypotension or renal dysfunction. Based on daily weight, symptom and sign monitoring, some patients can be taught to self-adjust their diuretic dose at home. There are several classes of diuretics: from loop diuretics (Furosemide, Bumetanide or Torasemide) to thiazides (Bendroflumethiazide, Metolazone or Indapamide) and potassium-sparing diuretics (Amiloride or Triamterene).

Common recommended dosages for the most common medications used to treat HF-REF patients are shown in Table 2.1. The ESC confirms that the usage rate of ACE-inhibitors, MRAs and beta-blockers is satisfactory. However, the number of patients treated with appropriate doses is still under the preferred level (Heart Failure Association of the ESC, 2010).

There are other pharmacological drugs with lower levels of evidence for HF treatments. Usually, they are prescribed if ACE inhibitors and ARBs are not tolerated or if there are persisting symptoms despite the previous treatments. Examples of these drugs are Ivabradine, which slows the heart rate in some patients; Digoxin, a digitalis glucoside; H-ISDN, a vasodilator; and PUFAs, a group of n-3 polyunsaturated fatty acids. Most of them have shown reasonable benefits of symptom reduction proving to be useful alternatives or additional therapies. Other types of treatments for HF-REF are still in research so enough evidence can be achieved for their secure recommendation.

Non-surgical treatment devices include wearable defibrillator vests or implantable monitors. Implantable cardioverter-defibrillators and cardiac resynchronization therapies are also possible treatments for HF-REF. For end-stage HF or other critical situations, surgery or even heart transplantation may be performed.

The guidelines from ESC (European Society of Cardiology, 2012) also present indications on drugs that should be avoided because they may cause harm to patients with HF-REF. Drugs prescribed for co-morbidities may interact with the ones used to treat HF. Consequently, correctly managing co-morbidities and their medication is very important. Among the contraindicated drugs for HF patients that can worsen HF there are tricyclic antidepressants for depression, non-steroidal anti-inflammatory drugs for arthritis, thiazolidinediones for diabetes or drugs for hypertension.

Table 2.1: Evidence-based doses of effective pharmacological drugs used for HF-REF (European Society of Cardiology, 2012).

	Starting dose (mg)	Target dose (mg)
ACE inhibitor		
Captopril[a]	6.25 t.i.d.	50 t.i.d.
Enalapril	2.5 b.i.d.	10–20 b.i.d.
Lisinopril[b]	2.5–5.0 o.d.	20–35 o.d.
Ramipril	2.5 o.d.	5 b.i.d.
Trandolapril[a]	0.5 o.d.	4 o.d.
Beta-blocker		
Bisoprolol	1.25 o.d.	10 o.d.
Carvedilol	3.125 b.i.d.	25–50 b.i.d.
Metoprolol succinate (CR/XL)	12.5/25 o.d.	200 o.d.
Nebivolol[c]	1.25 o.d.	10 o.d.
ARB		
Candesartan	4 or 8 o.d.	32 o.d.
Valsartan	40 b.i.d.	160 b.i.d.
Losartan[b,c]	50 o.d.	150 o.d.
MRA		
Eplerenone	25 o.d.	50 o.d.
Spironolactone	25 o.d.	25–50 o.d.

2.1.5. Living with Heart Failure

As Heart Failure is a chronic disease with no cure, a new lifestyle must be adopted by the patients for the rest of their lives. Patients are advised to perform regular aerobic exercise as well as some diet restrictions. They have to avoid excessive fluid intake, prevent malnutrition, keep a healthy weight, and follow a sodium restriction. Essential behaviours, from no smoking, to alcohol abstinence and prevention of depression, must be present. Besides these concerns, patients have daily to monitor weight, blood pressure and heart rate. Other occurrences like sleep disturbances should also be taken in account. HF patients are taught to be vigilant of their condition and know when they should notify the healthcare provider. Urgent alterations on patient condition include rapid weight gain (more than 2kg in 3 days), increasing shortness of breath and constant oedema (fluid retention in the body) (European Society of Cardiology, 2012).

2.2 Healthcare & Technology

Follow up and daily monitoring has always been an important part of healthcare, especially for chronic diseases like HF. It was seen that close home monitoring by a nurse practitioner can help diminishing readmission rates for HF patients (Naylor, et al., 2004). The quality of healthcare seems to be correlated to the support that patients are given at home. With the objective of helping these caregivers, basic methodologies and new technologies allow easier health monitoring and successful self-management of medication.

2.2.1 Basic Home-monitoring Tools

From glucometers to cardiac health monitors encompassing a variety of wearable or implantable sensors, many tools have surfaced for home-monitoring of vital signs and conditions. Here, only the tools for monitoring medication or treatment plans will be briefly explored.

Telemonitoring refers to the process of remotely monitoring patients who are not at the same location as the healthcare provider. Recent studies suggest that home telemonitoring appears to be particularly useful for patients with chronic heart failure, as the progressive deterioration of their clinical status and frequent re-hospitalisations constitute inevitable elements of the natural history of HF (Mortara, et al., 2009). Structured telephone support and telemonitoring can help reducing CHF-related hospitalisations and the risk of mortality. Phone sporadic monitoring has already shown to improve disease outcomes (Grancelli H, 2003). This practice can be incorporated in other healthcare procedures enhancing quality of life, reducing costs and promoting evidence-based prescribing. Even though there are many advantages, telemonitoring practices are still blocked by technical, behavioural, managerial and financial barriers (Jimison, et al., 2008).

Useful devices for managing medication include voice output devices. Voice prescription labels (VPL) or medication label readers are miniaturized electronic devices that offer audio information of prescriptions or medication (Allnatt, Engelhardt, Gao, & Mariano, 2001). They can take the form of pill boxes, medication dispensers, voice recorders or alarm watches. A physician or pharmacist has to record the information from the prescribed medication regimen. They are aimed especially for the elderly and visually impaired patients. The major advantage of these systems is their portability. Some of them have a medication reminder device incorporated. The overall objective of these technologies is to remind and prevent taking the wrong pill. ScripTalk (Raistrick, Phillip C; Raistrick, David; En-Vision America, 2006), Talking Rx (Millennium Compliance Corporation, 2004) and the MEMS 6 TrackCap (Aardex group, 2002) are examples of these devices. Nevertheless, their weakness lies in the fact that the device is not incorporated in something that the patient always carries with him. Furthermore, if the patient misses a dose, he still will not know what to do. Under a new approach, a recent patent presents Rx Zero (USA Patent No. US8303500, 2012), a non-pharmaceutical device to help prevent and monitor chronic diseases by managing, measuring, quantifying, monitoring and motivating appropriate lifestyle behaviours for the patient. Even so, the lack of understanding of the disease and treatment as well as the cost of assistive technologies is the most significant barrier to the use of assistant technologies.

2.2.2 Natural Language Generation Tools

Tailored hypermedia information can be easily consulted by patients at home. These systems use artificial intelligence and natural language generation (NLG) techniques to generate personalized media with hypertext containing basic factual information about a certain patient medical record, disease and treatment. Depending on the method and the disease, the patient can have simple and cheap access to the information through different media being a leaflet, a webpage, an online questionnaire, an interactive application, etc. Among many others, PIGLIT (Patient Information Generated by Loosely Intelligent Techniques) (Binsted & Jones, 1995), HealthDoc (Hirst, DiMarco, Hovy, & Parsons, 1997), OPADE (Carolis, et al., 1996) and Patient Advocate (Miksch, Cheng, & Hayes-Roth, 1996) are examples of these systems. The objective of these systems is to ensure that patients have the information required to understand and manage their condition. On the other hand, some effort must be spent to design the knowledge database with the general medical information that may be communicated to the patient. Another noticeable problem of these tools is that a recommended drug for a particular disease appears once in the knowledge database but it may be mentioned in textual documents using many different surface forms.

Artificial intelligence and NLG tools can be used on healthcare technologies on a more complex level: the conversational assistants. The Personal Medication Advisor fits within this class of systems that is still on an early stage of growth. Already developed or in-development conversational systems for home-monitoring or other medical applications are discussed in section *2.3.5*.

2.3 Conversational systems

Conversational assistants are computerized systems that communicate and interact with human users to help solving problems. The conversational paradigm implies understanding a verbal input, through speech recognition and context understanding, as well as verbalizing a response, using language generation and speech synthesis (Seneff, 2003). This way the assistant can be engaged in dialogue with the user. A conversational assistant should focus on practical dialogues, meaning that both interlocutors are cooperating to perform a certain task and achieve a certain goal. Practical dialogues can comprise tasks from planning to management, information retrieval or advice-giving (Allen, Ferguson, & Stent, 2001). Different types of conversational assistance differ on the tasks to be performed, their complexity and the degree of initiative taken by both interlocutors. According to the balance of initiative on the conversation, it is possible to have directed, mixed-initiative or multimodal dialogues (Seneff, 2003). While directed dialogue systems are already successfully commercialized, multimodal systems include action interaction and are starting to be developed. Mixed-initiative dialogue assistants are used for specific applications, usually staying as research prototypes.

A brief and generalized representation of the structure of a conversational assistant is represented on Figure 2.1. Conversational systems are complex combinations of different modules. A division of dialogue systems can be made according to the role of each group of components: interpretation, behaviour or generation (Ferguson, et al., 2006). Interpretation or natural language understanding (NLU) modules capture and understand what the user said. Behaviour or dialogue management components plan system actions by managing goals, intentions and reasoning. Generation modules construct the dialogue through content planning and surface realization followed by speech output.

When trying to assemble conversational technologies, there are four developing stages where attention should be focused, namely speech recognition, language understanding, language generation and speech synthesis (Cole, et al., 1996).

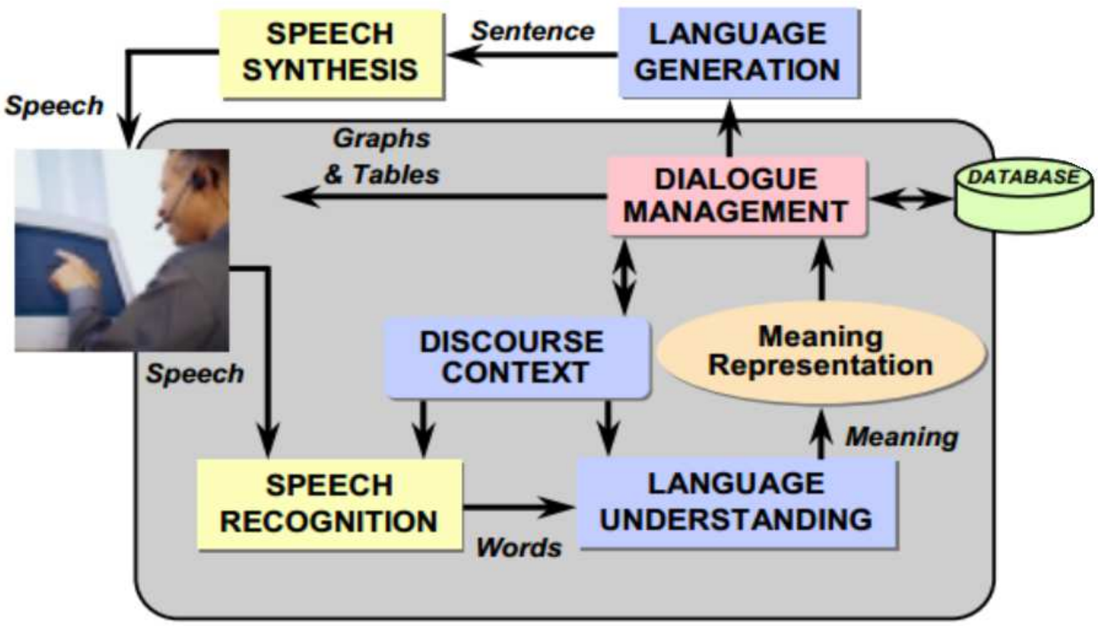


Figure 2.1: Schematics of a typical conversational system with the different functional modules. Source: (Seneff, 2003)

2.3.1. Speech Recognition

Automated speech recognition (ASR) components capture and process spoken input by converting the acoustic signal to a set of words. Recognized words will then be used as input for analysis. Most ASR systems use statistical methods, neural networks and/or Hidden Markov Models for acoustic modelling (Jelinek, 1997). An ASR system usually consists of front-end, an acoustic model, a language model and a decoder (Figure 2.2). The system will depend on the intended type of recognition that can be, for example, speaker dependent or independent, continuous or by isolated words. Spontaneous natural speech and its intrinsic variability can bring some difficulties to accurately recognize the speech. Variability in ASR lies on phonetic and acoustic dissimilarities as well as intra-speaker characteristics (age, emotional state, health, fluency or socio-linguistic background) (Benzeghiba, et al., 2007). Spoken input also differs significantly due to the false starts and filled pauses that characterize a natural speech. Even so, speech recognition systems have become much more robust in recent years with respect to both speaker and acoustical variability (Cole, et al., 1996). The system should be attentive for ungrammatical constructs, recognizing errors and modelling non-speech events and disfluencies. An extensive lookup on the available tools for recognition is made in *Chapter 3*.

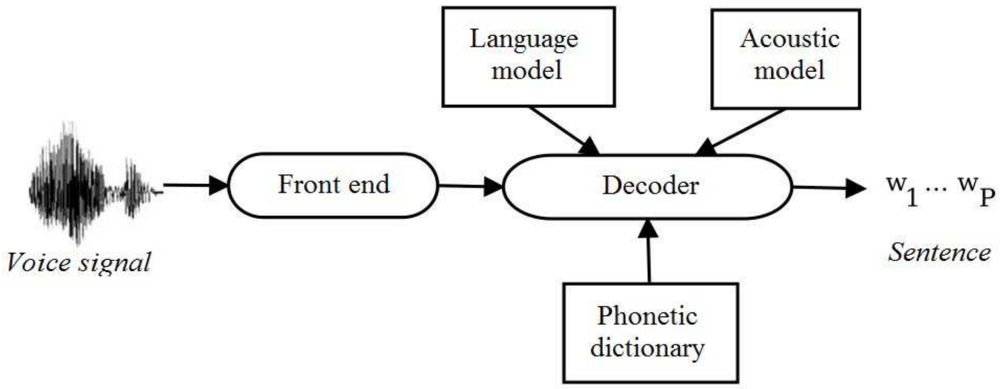


Figure 2.2: Overview of the structure of a typical ASR system. Source: (Neto N. , 2011)

2.3.2. Language Understanding

Between recognition and generation there is a NLU module to analyse and understand the acquired speech utterances. Encoded linguistic knowledge, like lexicon, grammar and semantics, helps producing a meaning representation or logical form through parsing and knowledge representation. The representation of the information acquired using adaptive models is of great importance, allowing it to be available afterwards for other applications (Cole, et al., 1996). Parsing establishes the syntactic organization as well as the semantic content of what was said. This is a hard task because subtle differences in phrasing and typical discourse constructions like ellipsis and pronominal or verbal referencing can lead to different interpretations. Ambiguities and ill-constructed sentences should be avoided by the system. Dialogue management tools are then responsible for clarification sub-dialogues, ambiguity dissolution and pragmatic considerations of what was said. The dialogue manager is the central unit for these tasks, generating and evaluating the hypotheses to the intended meaning of the user speech. After choosing the most possible intention of the user, the information is passed on to the next module presenting a hierarchy of active or pending goals and sub-goals. If the intention was not understood, the system should question the user. The dialogue manager should also be responsible to update dialogue context on each iteration so that the system can always be well fitted in the conversation (Traum & Larsson, 2003).

2.3.3. Language Generation

Natural language generation (NLG) refers to the process of automatically converting an abstract meaning representation into a string (Baptist, 2000). NLG aims to produce high-quality natural language text from computer internal representations of information.

2.3.3.1 Processes

There are several different approaches to language generation. Generation processes can be mainly categorized as non-linguistic (being statistical, canned-text or template-based systems) or as linguistic (being phrase-based, MTT-based or feature-based systems) (Baptist, 2000). The shallow simplicity of non-linguistic generators opposes to the sophistication of linguistic approaches that require much more planning and development. In general, linguistic methods are more robust and present high quality output. Linguistic generators entail simultaneous computational and linguistic knowledge relying on linguistic knowledge databases (LKB), this is, sets of files comprising grammar rules, lexical entries and rewrite rules, which specify the generation domain.

A statistical NLG system depends on training and probabilistic models that can be more or less complex (Ratnaparkhi, 2000). Although these systems are fast, they lack in accuracy and flexibility. Canned-text generators are the simplest but most wasteful systems. They only produce a static string of words without any amend, adequate for periodic event-specific generation (Cole, et al., 1996). More sophisticated, template-based systems can introduce minor alterations on generated sentences while a major part of the text is still static. Unlike previous approaches, they can perform multi-sentence generation if the text has a rather constrained structure (Reiter, NLG vs. templates, 1995).

Phrase-based systems focus on text planning by using generalized templates and replacing phrasal patterns by selected words. Their robustness works at sentence or discourse level (Hovy, Planning coherent multisentencial text, 1988). However, their weakness lies on the shortage of text plan libraries and on the complex phrasal interrelationships present at longer discourses, not fitting real scenarios. Meaning-text theory (MTT) based generators sequentially map semantic, syntactic, morphological and phonetic levels of representation according to a set of rules. The description of these relationships is made using lexical functions (Goldberg, Driedger, & Kittredge, 1994). With feature-based systems, each possible alternative of a language expression is analysed and categorized in a feature. As consequence, each sentence is characterized by a particular group of features. The final sentence is determined by incrementally match each inputted phrasal segment to the suitable feature. For these systems, the interrelationships are still difficult to manage and the number of features that can be identified must be controlled. There are no multi-sentence featured-based generators (Hovy, Language generation, 1996).

NLG was effusively studied between the late 1980s through the 1990s. Initially, simpler approaches like statistical or canned-text systems were preferred. However, over the years, there was acceptance of template-based and, later, of linguistic generators (Baptist, 2000). Integration of template-based and linguistic approaches would be suggested by Ehud Reiter (Reiter, NLG vs. templates, 1995), in 1995, opening a path to hybrid generation systems. Within a hybrid framework, encompassing both linguistic and non-linguistic aspects may present interesting advantages, allowing either to rapidly develop tools for simple domains or to use expert knowledge bases for more challenging domains (Baptist, 2000). The arrival of machine learning techniques after the mid-1990s has allowed further studies on linguistic approaches, mainly on sentence planning (Walker, Rambow, & Rogati, 2001), lexical selection (Bangalore & Rambow, Corpus-based lexical choice in natural language generation, 2000), word order (Bangalore & Rambow, Exploiting a probabilistic hierarchical model for generation, 2000) and grammatical relations (Corston-Oliver, 2000). Stochastic approaches and machine translation are bringing many advances to NLG (Chambers & Allen, 2004). With this, new hybrid methodologies appeared, namely the corpus-based stochastic generation where linguistic models are combined with statistical models derived from annotated corpora (Oh & Rudnicky, 2002).

2.3.3.2 Tasks

A modular approach to NLG is helpful so the developer can first be focused in managing and testing each of the three central tasks that comprise the generation process (Hovy, Language generation, 1996).

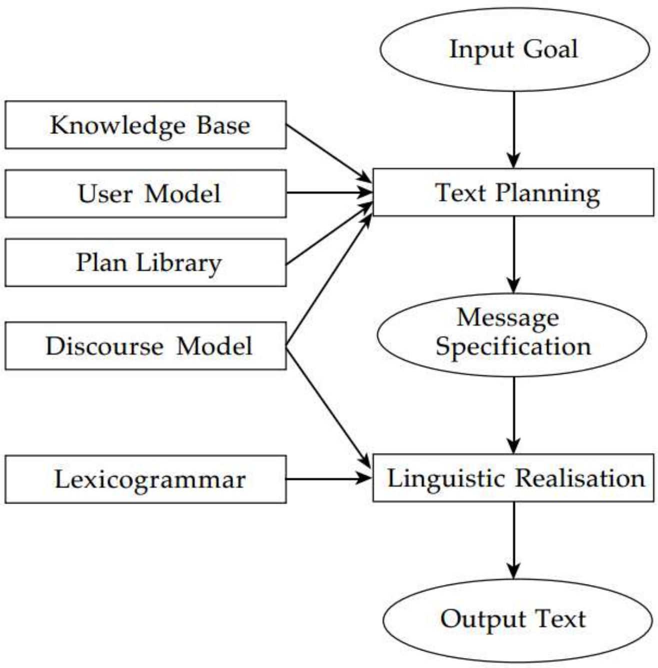


Figure 2.3: Overview of the tasks of a generation module. Source: (Dale, 1995)

Then, all the task responsible modules should be assembled and their communication assessed. Different models for the NLG system architecture and interconnection can be found (one example is presented on Figure 2.3). Although not completely independent, each generation task is briefly presented.

- *Text planning or content determination*

This first step defines global text structure, deciding *what to say*. The inputted information is mapped, the text content is established and the information organized accordingly. There are several approaches to text planning as the traditional top-down planners using Rhetorical Structure Theory (Hovy, Planning coherent multisentencial text, 1988) or intentional operators (Moore & Paris, 1993).

- *Sentence planning*

The information is structured and ordered into sentences and paragraphs using appropriate reference words as conjunctions and pronouns. Common tasks include pronoun specification, focus signalling, content aggregation to remove redundancies and preliminary phrase ordering. Lexicalization, the transition from concepts to lexical representations, can be included within this task. The lexical choice includes collocations and definition of lexical sub relations (Busemann, 1993). This module is thus seen as a mediator between deep generation (text planning) and surface realization because it limits simultaneously *what* can be said and *how* it can be said. Even so, a clear division between this and the other NLG modules is not always very visible (Smedt, Horacek, & Zock, 1996),

- *Surface realization or surface generation*

Using grammar knowledge, coherent and grammatically correct sentences are produced in formatted output text. The realization of the planned expressions, determines *how to say* the information according to morphological and phonological rules. Computational models try to mimic human utterances to approach the output to natural speech. A common approach is the stochastic surface realization: language models are built, candidate utterances are generated and scored, the best utterance is selected and finally the missing slots are filled with the relevant input information (Oh & Rudnicky, 2002).

2.3.3.3 Demands of language generation

Developing a NLG system requires good structured lexicons and grammars in the language(s) that the system will use; knowledge bases oriented to the domain-specific application that is being developed and consolidated text planning libraries. Appropriate text-to-speech and speech-to-text systems would be necessary to help integrating the NLG module on a conversational system. Naturalistic speech generation with the inherent disfluencies, interruptions or overlaps is still distant due to the unsolved difficulties related to discourse and interpersonal speech variability.

Until now, only general purpose single-sentence or limited purpose multi-sentence generators were developed (Hovy, Language generation, 1996). A multimodal general purpose language generator has not been build due to the unsolved problems that appear when using large lexicons, unclear contexts, unconventional discourse structures, large-sized domains or systems with great expressive potential that can say the same thing in many ways. In this way, there is still a lot of work to be done on developing generators with the desirable coherence and consistency, producing texts of several paragraphs functional on real-world domains.

Challenges arise when trying to build multilanguage systems. As the major amount of NLG research is done using English language, generation in Portuguese is still poorly explored. Furthermore, European Portuguese has a complex verbal structure with auxiliary verbs (Baptista, Mamede, & Gomes, 2010). For now, more literature can be found on Portuguese language recognition, parsing and processing (Meinedo, Caseiro, Neto, & Trancoso, 2003) (Silva, Branco, Castro, & Reis, 2010) rather than on Portuguese NLG.

2.3.4. Speech Synthesis

Speech synthesis is the automatic generation of speech waveforms. Spoken output of NLG sentences can be produced by text-to-speech (TTS) systems (Figure 2.4). The first speech synthesis systems were hardware-based. With the increase of computational power, today’s synthesizers are mostly software-based, allowing more functionalities, more flexibility and frequent updates (Lemmetty, 1999). In the majority of TTS systems, the first step is text analysis, or high-level synthesis, where a grapheme-to-phoneme conversion is done. Character strings are pre-processed and converted into phonetic representations. These representations are typically a string of phonemes that can also contain information about phasing and intonation. On a second phase, called low-level synthesis, phonemic representations are converted to waveforms that can be outputted as sound. The majority of actual systems are composed of both high and low synthesis components.

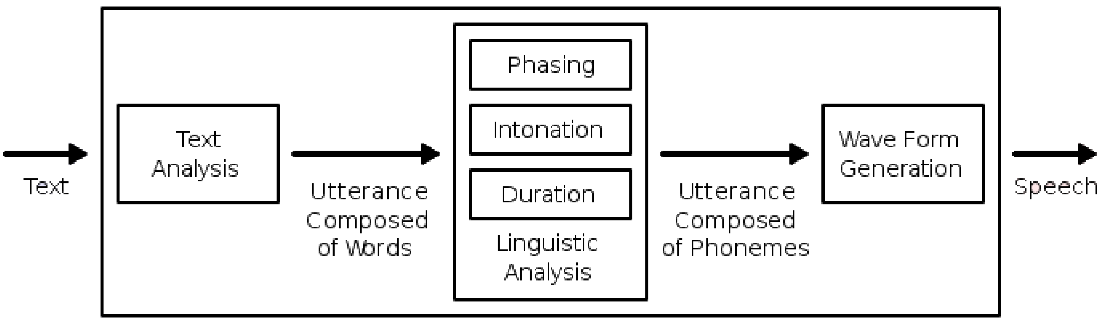


Figure 2.4: Overview of the structure of a typical TTS system. Source: (Vasavi & Sravya, 2010)

There are several methodologies for generating synthetic speech waveforms: from concatenative to articulatory, formant or Hidden Markov Model based synthesis. (Lemmetty,

1999) Each TTS system uses one of these methods according to its own strengths and weaknesses and the intended use of the system. Concatenative TTS systems require pre-recorded units and sound libraries based on human speech recordings. With this, they provide a synthesized speech with a more natural sound. This naturalness implies the transcription and alignment of large corpora, what can be time-consuming and demand a profound knowledge of phonetics to accurately align temporally the transcription labels (Carvalho, Oliveira, Viana, & Trancoso, 1998). Formant TTS systems are rule-based requiring less resources. Although the voice quality is lower and less natural, the speech is clear and many languages can be provided with a small size system. This may also make these systems faster than concatenative TTS. Modern speech synthesis technologies involve complex methods and algorithms. Among recent approaches to speech synthesis, Hidden Markov Models and Neural networks based methodologies are very used (Schroeder, 1993).

TTS engines also differ with different languages, dialects and specialized vocabularies. High quality speech synthesis systems should maximize naturalness and intelligibility (Lemmetty, 1999) (Thakur, Chettr, & Shah, 2012). The current speech synthesis systems can already artificially produce highly intelligible human speech. However, achieving an optimal naturalness with synthesized speech can still be a problem (Cole, et al., 1996). An overview of existent TTS tools is available in *Chapter 3*.

2.3.5. Current Conversational Systems

A majority of the existing conversational assistants is available for profitable applications. They include flight booking assistants, weather informers, restaurant or hotel guides, calendar managers, city navigators, traffic reporters, sports updaters, crisis supervisors, among others (Seneff, 2003). After several years of development in MIT, *Genesis II* is an example of a hybrid conversational assistant presenting a framework for several languages and several domain applications: from *JUPITER* in weather domain to *MERCURY* in flight reservation, ORION in off-line task delegation or *VOYAGER* in navigation assistance (Baptist, 2000). *Amalgam* was developed by Microsoft focusing on sentence realization and using machine-learned and knowledge-engineered modules in machine translation contexts (Gamon, Ringger, & Corston-Oliver, 2002). Other systems include the *IBM Flight Information System*, (Axelrod, 2000) the *CMU's statistical generator* (Jin, 2003) and a collaborative multimodal system for rescue emergency scenarios *TRIPS* (Allen, et al., 2001). These systems are usually based in English, German, Japanese and/or French Language generation. Research on European Portuguese NLG and conversational systems is still limited. *Meteo* (Cardoso, Flores, Langlois, & Neto, 2002), a telephone-based conversational system for weather forecast and the spoken dialogue virtual butler of the *Interactive Home of the Future* project (Neto & Cassaca, 2004) are examples of the few conversational systems developed in European Portuguese.

Medicine related systems started to appear on the last decade. An example of surgery applications for NLG and intelligent assistance is *GALEN-IN-USE system* for the collaborative construction and maintenance of surgical procedure classifications (Wagner, Rogers, Baud, &

Scherrer, 1999). Conversational assistants using NLG for healthcare are still in development like the Chester (Ferguson, et al., 2006) and *CARDIAC* (Galescu, Allen, Ferguson, Swift, & Quinn, 2009) systems. The *CARDIAC* is an English spoken dialogue system that conducts health monitoring interviews using natural language for chronic heart failure patient healthcare. Rather than medication assistance, it collects information about the current state of patient's health from objective (weight) to subjective (pain, fatigue, exercise) conditions. *CARDIAC* can be useful for patient self-care, for monitoring trends and generating reports for healthcare providers. In a way, it shares part of its motivations with the Personal Medication Advisor.

New and more complex approaches have appeared, using NLG and speech along with sensing, acting and gesture recognition to assemble multimode robot systems in what is called an Embodied or Situated Conversational Assistant (Mavridis, 2007). They include robotic lab assistants, household helpers and affective robotic computers. They may portray the future of conversational assistant technologies.

2.4 Summary

Over this chapter, the motivation and problem behind this project was further contextualized by characterizing the world-wide Heart Failure healthcare scenario. Heart Failure is a disease with increasing incidence associated with high hospitalization costs, high mortality and morbidity. These patients need to follow strict medication regimens, composed of ACE inhibitors, ARBs, MRAs and beta-blockers, to reassure their survival and reduce rehospitalisation. Daily home-monitoring and medication compliance are extremely important for HF patients. There are several home-monitoring tools they can use.

Conversational assistants can be encompassed within a HF home-healthcare environment. The main tasks of a conversational system are to understand what the user says, to plan an appropriate response and to express a response to the user through generated natural language. Thus, these systems usually have a modular structure, with modules for speech recognition, natural language understanding, dialog management, natural language generation and speech synthesis.

Chapter 3

Tools and Resources

This chapter identifies the resources and tools that were exploited along the different steps and iterations of the Personal Medication Advisor. Although some of these tools are not directly used by the final prototype, they played an important role during the development process.

With this, development platforms are pointed up, followed by the resources and tools that were necessary to this project. From speech technologies to the linguistic tools used, an explanation of their characteristics, purpose and the rationale for their choice is given.

3.1. Development platforms

During an initial phase, the medical assistant was developed for personal computers, using Java in the Eclipse environment. Java was chosen as the programming foundation of PMA because it is simple, portable, free-ware, allows easy integration of external packages and supports modular programming. Even so, the most important reason was to facilitate the integration of the system with Smartphone technologies. Consequently, at a later stage, PMA was rebuilt as an Android application, using Java and the Eclipse IDE bundled with Android Development Tools (ADT).

3.2. Resources

3.2.1 Ontologies

The required medical information regarding medication posology, dosage, indications and adverse reactions, among others, was retrieved from an ontology. An ontology can be defined as 'a formal, explicit specification of a shared conceptualization' (Studer, Benjamins, & Fensel, 1998), this is, a model of a certain system that takes the form of a definition of properties of its important entities and relationships, being specified in an unambiguous language and processable by both machines and humans (Staab & Studer, 2009). With this, an ontology is characterized by well-defined syntax and semantics, efficient reasoning support, sufficient expressive power and convenience of expression. The final objective is to provide a meaningful shared vocabulary by presenting customized concepts and properties along with their definitions and constrains, so that it can be used for domain specific communication between people and information technology (IT) systems. Ontologies are usually designed through XML-based or web semantic languages such as resource description framework (RDF) or the expressive Web ontology language (OWL). This type of resources is now essential to many applications such as scientific knowledge portals, information management, electronic commerce or semantic web services.

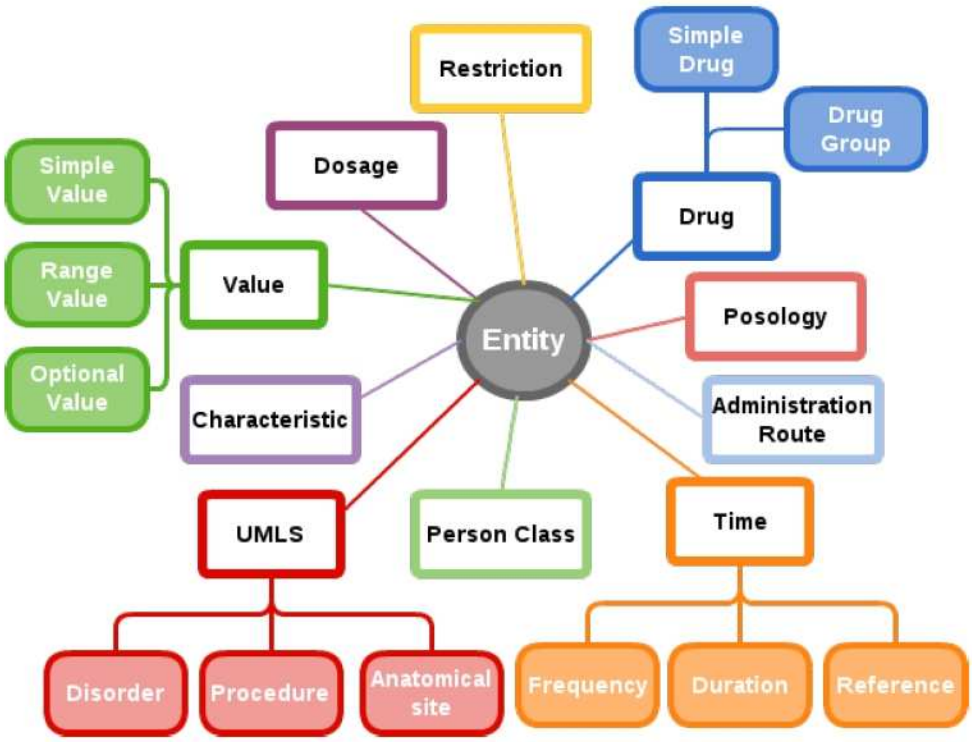


Figure 3.1: Tree of entities, types and subtypes for the *PharmInx* ontology. Source: (Aguiar, 2012)

The OWL based ontology used as the knowledge resource of PMA was created through the *PharmInx* (Pharmacological Information Extraction) system under the context of another project at Fraunhofer (Aguiar, 2012). This system arose from the need of having a methodology to design a complete, well-defined and reliable medical vocabulary that would be shared between computational systems, healthcare application users and professionals. Hereby, *PharmInx* comprises automatic extracting and modelling of information from Portuguese medical therapeutic records and leaflets. This tool was mainly developed to extract posology information using extensible knowledge models, structuring entities and their respective relations extracted from pharmacological texts. A representation of the entities and types of the

PharmInx ontology can be seen in Figure 3.1. Several methods are used for the extraction and structuring of information, from regular expressions to external information sources' lookup or machine learning techniques. Extraction from free-form text written in Portuguese and automatic population of the ontology with the information extracted were achieved with high precision when extracting posology. The *PharmInx* ontology was thus designed and maintained resorting to the Protégé framework, whereas ontology's population performed with JENA API. The PMA will also make use of these two tools, presented in detail on the next section. Even so, *PharmInx* was not able to extract all kinds of pharmacological information. This could have confined the ontology that was used and consequently constrained the information that PMA could use.

An organized and structured representation of medical information is thus available in European Portuguese. The Personal Medication Advisor used this knowledge representation model, restricting it to the heart failure pharmacology domain. The fact that PMA is based on ontology knowledge is a strong uniqueness of this project.

3.2.2 Linguistic resources

As a conversational system, PMA is very dependent on linguistic resources like European Portuguese LKBs and domain specific lexicons. *Linguateca* (Monteiro, et al., 2008) is a resource centre for computational processing of the Portuguese language which provides useful materials for linguistic analysis and processing. Along the development process, several language resources were tried out, namely some domain independent Portuguese dictionaries and lexicons. These resources were needed for language processing and generation. As the system was adapted to the Android platform, the linguistic resources were modified and restricted to a domain specific vocabulary so that the system required less memory and had a faster performance.

3.3. Ontology tools

3.3.1 Jena

Jena is a Java programming framework for ontology applications development, handling RDF, OWL and SPARQL (SPARQL protocol and RDF query language) (The Apache Software Foundation). It is an open-source project started in 2000 by HP Labs and maintained by The Apache Software Foundation.

The *Jena* application programmatic interface (API) allowed reading and modelling the ontology as well as searching and retrieving the desired information through SPARQL queries. SPARQL is a query language to retrieve and manipulate data stored in the RDF format. When used along with *Jena* API, these queries were very helpful since they allowed retrieving from the ontology the specific information to answer a certain question asked by the user.

3.3.2 Protégé

Protégé is another Java based tool, useful for working with ontologies. It is an open-source project by the Center for Biomedical Informatics Research at Stanford University (Stanford Center for Biomedical Informatics Research). It works as a platform for ontology creation, editing and visualization. It has a user friendly interface and it also supports a variety of ontology formats such as RDF, OWL and XML Schema. *Protégé* is an extensible and customizable framework that can be easily integrated with other systems.

As the ontology that was used was already created and modelled, *Protégé* was used mostly for visualization of ontology individuals and their relationships. An example of an ontology individual and its properties in *Protégé* is shown in Figure 3.2.

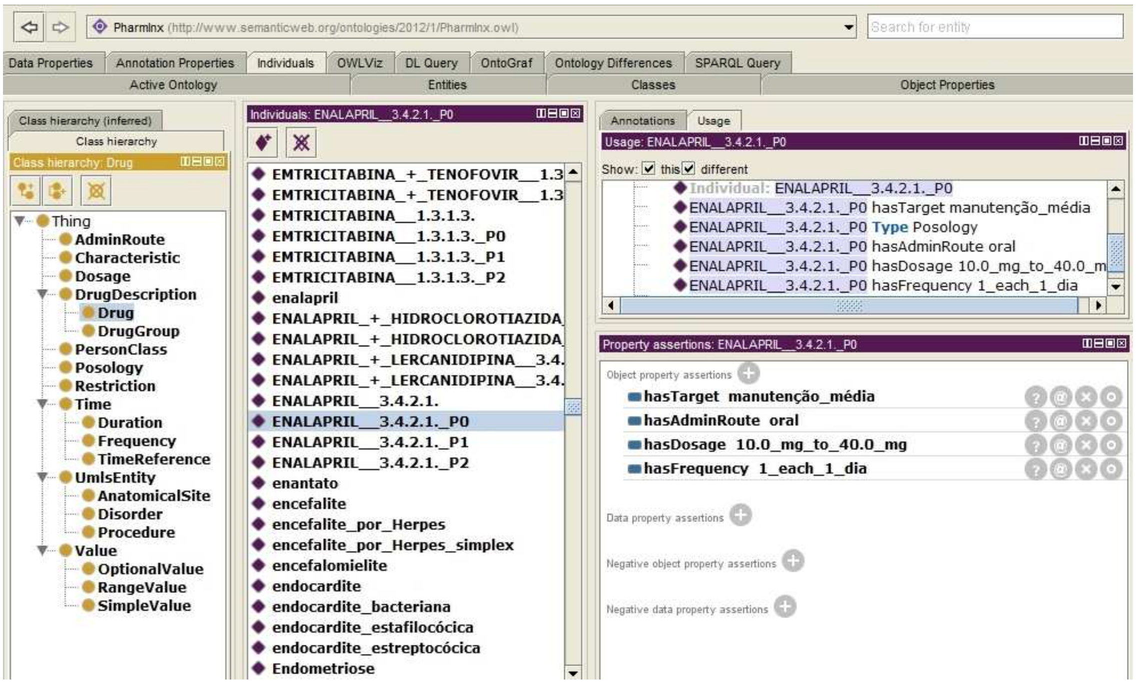


Figure 3.2: Individuals view for the *PharmInx* ontology on *Protégé*.

3.4. Language tools

Part-of-speech taggers and phrase chunkers are crucial for natural language processing. They divide phrases and annotate, for each word in the text or sentence, its corresponding part-of-speech, this is, its lexical category. *JTextPro* and *TreeTagger* are open-source linguistic tools that were used for language processing in the PMA project.

3.4.1 JTextPro

One part-of-speech tagger for English can be *JTextPro* (Phan, 2006), a Java based toolkit for text processing, developed in Tohoku University. It can effectively perform word tokenization, phrase chunking and part-of-speech tagging for English texts. This tool was used for the initial stage of the project that used the English language as it will be later explained. It presents a good performance, with successful tagging.

3.4.2 TreeTagger

TreeTagger is a language independent part-of-speech tagger that has been successfully used to tag many languages including English and Portuguese. For each word, it identifies the word class (part-of-speech) and the lemma. *TreeTagger* uses decision trees, instead of the popular Markov Models, achieving good precision rates. It was developed by Helmut Schmid for the TC project at the Institute for Computational Linguistics of the University of Stuttgart (Schmid, n.d.). For better results, each language should work with a specific parameter file containing a trained corpus. European Portuguese parameter file was developed by Pablo Gamallo of University of Santiago de Compostela (Otero, 2005). Tagging for Portuguese achieves good results with *TreeTagger*.

3.5. Speech tools

Voice technologies consist of speech synthesizers and recognizers, being key tools for natural spoken human-computer communication. Both synthesis and recognition modules are not the main focus of this work but rather tools to allow the development of the conversational assistant and implementation of Natural Language Understanding and Generation techniques. Nevertheless, recognition and synthesis platforms had to be carefully chosen so that integration with PMA's NLU and NLG modules could be successfully achieved. As these tools presented some constrains to system development, early selection of these components was recommended. It is also important to note that quality and accuracy of both synthesis and recognition are not the most important factors when choosing the speech tools to use for this project. With this, the proposed requisites for selecting speech tools were:

- Have an European Portuguese language and acoustic models;
- Be written in Java or support Java Speech API by Sun Microsystems Laboratories;
- Be free-ware;
- Be portable;
- Allow specialized medical vocabulary;
- Have acceptable quality and accuracy.

Speech synthesis research started even before the digital revolution, when Homer Dudley presented Voder and Vocoder, in 1939 and 1940, at Bell Telephone Laboratories, being the initial attempts to electronically mimic human speech (Dudley, 1939). The first automated speech recognition tools were also developed at Bell Laboratories (the "Audrey" system) later in the 50s (Davis, Biddulph, & Balashek, 1952), followed by RCA Laboratories (Olson &

Belar, 1956) and MIT Lincoln Lab (Forgie & Forgie, 1959) systems. But it was only during the 80s that speech research prospered and, as technical limitations diminished, new opportunities were brought to speech technologies.

3.5.1 TTS systems

There is a long list of TTS systems, from open source to commercial ones, from discontinued to in-development products, from commonly used to unheard of systems. Some examples are *Ximera* by ATR Interpreting Telecommunications Labs, *IVONA* by IVONA Software, *DECtalk* by Digital Equipment Corporation, *Infovox* by Acapela group, *Natural Voices* by AT&T labs, *ORATOR* by Telecordia, *CyberTalk* by Panasonic, *Vocalizer* by Nuance, *HERON* by University of Aveiro and much more. The synthesizers *FreeTTS* by Sun Microsystems Laboratories, *Festival* by CSTR University of Edinburgh, *FurbSpeech* by Germano Fronza and Israel Medeiros and *eSpeak* by Jonathan Duddington were selected as possible candidates to integrate in PMA. These systems were evaluated according to the established requirements.

If the PMA was to be set with the English language, *Festival* or *FreeTTS* would be a good choice. *Festival* is one of the most complete free-ware TTS. It offers diphone concatenation synthesis through several APIs, being a general framework for building speech synthesis systems (Black, Taylor, & Caley, The Festival speech synthesis system: system documentation, 2001). This synthesizer is written in C++ and supports Java Speech API (JSAPI). It is a good TTS system although it can be a bit slow due to its size. *FreeTTS* (Sun Microsystems Inc., 2005) is another TTS system that can be used for English. It is written in Java and was derived from both *Festival* and *Flite*. The project Personal Health Management Assistant from University of Rochester is similar to the PMA and used *FreeTTS* to generate speech (Ferguson, Personal Health Management Assistant, 2007). *eSpeak* (Free Software Foundation Inc, 2007) also seemed to be a good system, with the advantage that it already presents a model for the Portuguese language. However, it is written in C and does not support JSAPI. *eSpeak* uses formant synthesis while *FreeTTS* and *Festival* use concatenative synthesis and generate a more natural speech.

During the early development of PMA, that used using American English as language, synthesis was performed with *FreeTTS*. Speech output was fairly good and robust, although a bit slow sometimes.

For European Portuguese, there are a few synthesizers available but they are mostly commercial products. As we want to develop a freeware-based conversational system in European Portuguese, using *FreeTTS* or *Festival* would require building a new model with *Festvox*. There were no freely available and robust acoustic and language models for European Portuguese to integrate with the synthesizer. Models and voices for Portuguese could be generated using *FestVox* or other voice building tool. This would be another great challenge, but beyond the scope of PMA. Synthetic voice building is time-spending, calls for processing power and requires recording a fairly big sound database. To obtain a good voice model, training techniques for intonation, duration modeling and letter-to-sound rules are also necessary (Black & Lenzo, Building Synthetic Voices, 2007).

Synthesis in PMA was then achieved by conjugating *FurbSpeech*, a TTS API developed for Brazilian Portuguese (Fronza & Medeiros, 2009), with MBROLA voices (Dutoit, Pagel, Pierret, Bataille, & Vreken, 1996). Both Brazilian and European Portuguese MBROLA voices were used. The speech generated with each one of the three available Brazilian Portuguese voices was better and more understandable than the European one. The European Portuguese voice used was the only MBROLA voice available for this language and it lacks in naturalness, having particular problems in the synthesis of phonemes that contain 'nh', 'lh' or 'j'. Furthermore, with both Portuguese voices there are still some difficulties when synthesizing words with accentuation or special characters like 'ç'.

3.5.2 ASR systems

There are less ASR than TTS systems. Some recognition tools include: *Via Voice* by IBM, *Dragon Naturally Speaking* or *Nuance Recognizer* by Nuance, *Julius* by Nagoya Institute of Technology, *Windows Speech recognition* technologies by Microsoft, *Tazti* by Voice tech group, *AUDIMUS* by INESC-id and Natural language recognition tools by Simplis.

A big difficulty of the PMA project has been the choice of recognition tools for Portuguese. Recognition in European Portuguese is still undeveloped and very new. An ASR tool that could fulfil the requirements was CMU*Sphinx* (Carnegie Mellon University, n.d.). *Sphinx4* is Java written and allows building and incorporating language models. It is a flexible system. It has been used for several applications including the Personal Health Management Assistant from University of Rochester (Ferguson, Personal Health Management Assistant, 2007). The early version of PMA used *Sphinx* to recognize in English, having a predefined and auxiliary Java Speech Grammar Format (JSGF) that defined possible user utterances to be recognized. *Sphinx* recognition worked very well for American English. The use of an appropriate model should allow adapting this system for recognition in Portuguese. However, there are no free acoustic models or grammars for European Portuguese. With this, other alternatives were explored. One solution could encompass using *JlaPSAPI*, a Brazilian Portuguese specification of JSAPI for recognition with Java. This API is still under development, very user unfriendly and needs several external packages and installations. Other solution was combining *Sphinx* with a PMA specific grammar and existing Brazilian Portuguese dictionaries and language models. Due to the lack of acoustic training and the difference between European and Brazilian Portuguese accents, the accuracy of recognition seemed insufficient for this solution. The problem of ASR for European Portuguese could only be surpassed when the PMA project moved to the Android platform and made use of android speech tools.

3.5.3 Combined systems

There are systems were both TTS and ASR systems are embedded and combined: from the successful *Google Now* by Google or *Siri* by Apple to *LumeVox speech engine* by Lumevox or *Loquendo TTS and ASR* by Nuance. For example, *Zanzibar openIVR* by SpokenTech uses *FreeTTS* and *Sphinx4* and goes to the encounter of the tools chosen above. Nevertheless, most of these products are only commercially available for user-end purposes and not for developers. Among them, *iSpeech* technologies (iSpeech, n.d.) have demo versions that are available for Java and mobile environments and were tested with PMA. The synthesis in European Portuguese has very high quality and is very accurate. Unfortunately, speech recognition on the demo version is limited and synthesized speech sometimes includes a copyright marking in Java developing environments.

It was only during the latest phase of development, when PMA moved to the Android implementation, that the best arrangement of speech recognition and synthesis tools was found. Android speech recognition packages, *Google Voice* (Google, n.d.), were used as they worked better than *iSpeech* Android recognition. On the other way, *iSpeech* Android synthesis had more quality than Android speech synthesis and thus was preferred.

3.6. Summary

The platforms essential to the development of the project were described in this chapter. In summary, the Eclipse IDE bundle with ADT was the final platform of choice. The resources used included a medical domain specific ontology, obtained through the systematic processing of Portuguese medical leaflets, as well as European Portuguese languages resources such as a dictionary and a domain specific lexicon.

The tools used comprise Java frameworks for ontology modelling, visualization and questioning, in addition to language processing packages for part-of-speech tagging and phrase chunking. Among the several speech technologies that were assessed, there was a special focus in *CMUSphinx* and *Google Voice* for ASR, as well as *FreeTTS*, *FurbSpeech* and *iSpeech* for TTS.

Chapter 4

The Personal Medication Advisor

The Personal Medication Advisor was envisioned to be developed within an iterative patient centred approach for personal healthcare. During the whole process of development, the system aimed for consistency, simplicity and full applicability on home health environments, namely for heart failure care.

This chapter describes each step of implementation and development, along with an explanation of system architecture. First, a general overview of PMA's features and performance is given. Then, the final architecture of PMA, along with descriptions of module tasks, is clarified. Main features and guidelines of system architecture are shared by the different PMA prototypes during the different stages of the project. For this reason, the distinction between the different prototypes and stages is only given afterwards.

4.1. General overview

Before a detailed analysis of system architecture and of the different stages of development, a brief overview of system flow and functionalities of the system is needed.

PMA engages the interaction with the user by generating a greeting according to the time of the day (morning, afternoon or night) and then waits for a user utterance. Due to the complexity of coherent and complete conversational systems, the range of possible user utterances had to be narrowed. Ideally, the user should be free to say whatever he wants independently of the words used or the syntactic structure chosen. However, the nature of this project does not allow PMA to have such a wide flexibility. With this, the user can interact with the system with somewhat rigid and limited sentences. The user should question the system using utterances similar to the following ones:

- "*Qual é a dose de <nome medicamento> receitada?*"
  (What is the prescribed dosage of <medicine name>?)

- "*Qual é a frequência de* <nome medicamento> *receitada*?"

  (What is the prescribed frequency of <*medicine name*>*?*)
- "*Qual é a via para tomar* <nome medicamento>?"

  (What is the route to take <*medicine name*>*?*)
- "*Existem restrições para tomar* <nome medicamento>?"

  (Are there any restrictions to take <*medicine name*>*?*)
- "*Quais são as posologias existentes de* <nome medicamento>?"

  (What are the posologies of <*medicine name*>*?*)
- "*Quais são as reações adversas de* <nome medicamento>?"

  (What are the adverse reactions of <*medicine name*>*?*)
- "*Porque razão estou a tomar* <nome medicamento>?"

  (Why am I taking <*medicine name*>*?*)
- "*Quais são os medicamentos que o médico receitou*?"

  (What are the medicines that the doctor prescribed?)
- "*Posso tomar* <nome medicamento>?"

  (Can I take <*medicine name*>*?*)

These questions will be interpreted by the system and an appropriate answer will be given, making use of the information retrieved from PMA resources. For example, if the patients asks: "*Quais são os medicamentos que o médico receitou*?" (What are the medicines that the doctor prescribed?), PMA would list the names of the medicines that were on the prescription, along with a counting for the total number of medicines prescribed. When the user utterance is invalid or the system does not understand what was said, PMA will retaliate with a random set phrase that requests the user to repeat or reformulate the question.

The system can also promptly react to some basic words present in a normal conversation: "*sim" (yes),* "*não*" (no), "*ajuda-me*" (help me), "*112*", "*adeus*" (goodbye) or "*obrigado/a*" (thanks you). These interactions are not dependent of PMA resources as the previous questions. For example, "*sim"* (yes) or "*não*" (no) are useful for the user to answer some questions that PMA uses for confirmation or inquiry. As "*ajuda-me*" (help me) will trigger a brief explanation on the PMA system, "*112*" will make the system deal an emergency call to the national number of emergency 112. If the user wants to close the assistant, he should say "*adeus*" (goodbye). A few courtesy phrases are also triggered if "*obrigado/a*" (thank you) is said, giving a more natural feeling to the interaction.

The PMA has access to user's medical prescriptions. This way, information can be more reliable and tailored for each patient. The prescription must have the list of prescribed medicines along with the prescribed posology. For testing PMA prototypes, a medical prescription was simulated. Another feature of PMA includes the production of a medical report, resulting from an inquiry to the user, that allows the healthcare staff that is responsible for that patient to be informed when the he does not feel well or has a critical HF symptom.

A general dialogue task specification, in the form of the task tree presented in Figure 4.1, encodes the system's behaviour and allows a first insight into system architecture. Each system feature will be explained with more detail in the next section.

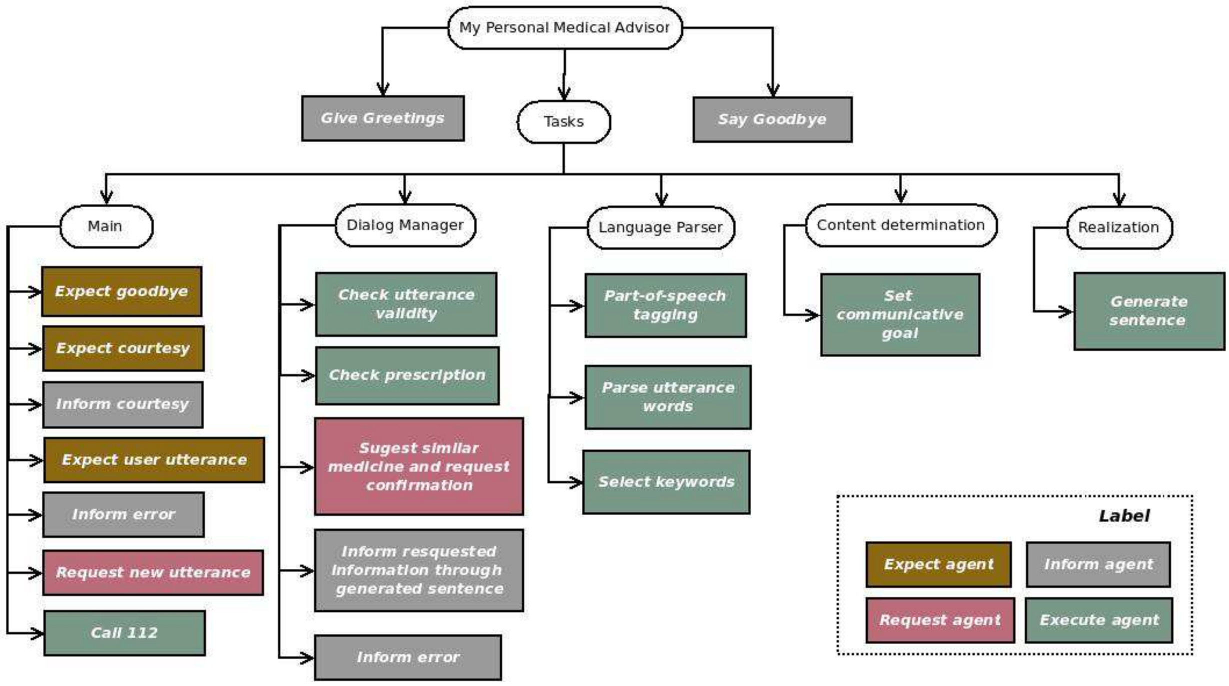


Figure 4.1: Task tree of PMA.

4.2. Architecture

The PMA was developed envisioning a modular architecture, this is, a top-down design. In this design strategy, the system consists of small and autonomous modules, each performing different functions but altogether working as single structure. This makes the system easier for debugging and for latter updating or modification. Having a modular structure is important for a conversational assistant, where different tasks are being carried and a specific module could be reused in other systems (Mellish & Evans, 2004).

The three essential modules that comprise the PMA were established as the language parser, the dialog manager and the language generator. Both speech recognition and synthesis modules allow these crucial components to directly interact with the user, using pre-existent speech tools. With this, they are integrated with the remaining components, making a total of five modules that compose the PMA system. A schematics overview of the system is shown on Figure 4.2. Initially this work was meant to have a special focus on the development of the language generation module. However, functional NLU and managing modules are indispensable for a system that tries to come nearer to a conversational system.

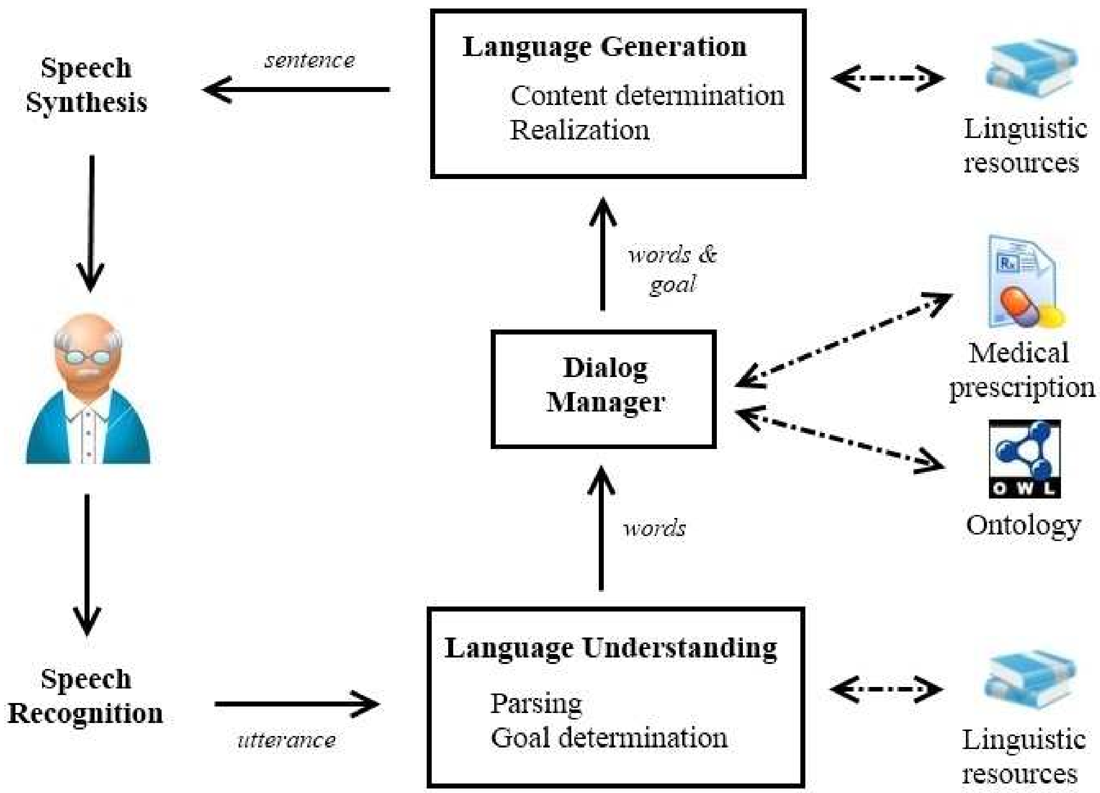


Figure 4.2: Schematics showing the flow between different modules of PMA and its resources.

4.2.1 *Language Parser*

The language parser is responsible for performing a rudimentary processing and segmentation of what was perceived by the recognition module. The whole process can be seen as a Natural Language Processing and Understanding procedure.

The initial phases of project development integrated external tools for language processing, such as the text taggers *JTextPro* and *TreeTagger*. However, due to the need of developing PMA on the Android platform and because there are no tools to perform word tagging in Portuguese for Android, it was necessary to rebuild this module. A simple routine that tokenizes and then categorizes each token of the sentence was implemented.

The first step is tokenization, this is, breaking the recognized text into meaningful elements (tokens) such as sentences and words. For this, a simple tokenizer routine was employed, making use of the *BreakIterator* package in Java. Afterwards, token list is parsed by a phrase chunker and a part-of-speech tagger. This syntactic analysis is required so that each sentence is segmented and each word is annotated with the corresponding word class and lemma. The parsing process identifies all the names and verbs on user utterances in order to spot keywords, extract the necessary information and infer the intention of the user. This parsing process relies on a grammar and a lexicon which can have a generalized range of words or can be domain-specific. The final prototype uses of a domain-specific lexicon to make computation faster and lighter. Furthermore, a more complex sentence processing method was not demanded by the current system.

In summary, the Language parser or NLU module starts to understand what the user says. The processed information will then be passed to the dialog manager.

4.2.2 Dialog Manager

The dialog manager is responsible for goal analysis and information retrieval. When uttered words are tagged and keywords identified, the system tries to bring out the intention of what the user said and, accordingly, delineate a goal for what should be communicated in response to the user. A set of communicative goals were defined for PMA: EXISTENCE, INFORMATION, LIST, PERMISSION, REASON and UNKNOWN. These goals are used in an attempt of question classification. By analysing utterances, the system tries to include each of the user's questions inside one of the mentioned attributes. As user utterances must always follow a predefined structure, it was settled that the keyword that establishes the goal, and thus reveals what the user requests from the system, is the first word of the utterance, in this case being an interrogative pronoun or a verb. This way, system performance is simplified and it is enough to associate a goal to each of the distinct intentions behind possible user utterances. For example, when the user questions the system "*Qual é …?*" (What is …), he must want to retrieve information related to a specific topic and the communicative goal is defined as INFORMATION. If the questions goes as "*Quais são…?*" (What are …), the user would want to see listed the attributes of a certain entity and the goal changes to LIST. If "*Porque ...?*" (Why …?) is asked, the user wants a justification and reasoning, and the goal is set to REASON. As for a utterance starting with "*Posso* …?" (Can I …?), the user would like to receive permission for something, being PERMISSION defined as the goal. The similar happens for the goal EXISTENCE. UNKNOWN is used when the goal could not be detected. Finally, when the goal is defined, the system acts according to the type of response that the user is expecting.

Afterwards, depending on the goal, the dialog manager tries to identify, on the list of the tokens identified as nouns, a known medicine name and/or one of the featured property names. It also checks if there are verbs in the phrase. If yes, the first verb is also selected as a keyword.

If the goal is defined as INFORMATION, LIST, EXISTENCE or REASON, the dialog manager initially looks for the medicine name on the ontology through specific SPARQL queries. After that, it will also check the user's prescription to know if the uttered medicine was prescribed by the doctor. If that medication is not prescribed or if that medicine name is not valid, the dialog manager performs a Levenshtein Distance iterative routine to assess the similarity between two strings, the uttered medicine name and each name of the prescribed medicines. The Levenshtein Distance algorithm measures the edit distance, this is, the minimum number of single-character edits, such as deletion, insertion, and substitution, necessary to change one word into the other. The medicine name that is on the prescription, and showed more similarity with the uttered medicine name, is then suggested to the user: "*Não quererá dizer* <*nome semelhante de medicamento*>?" (Don't you mean <*similar medicine name*>?). The user will have to confirm the judgment of the dialog manager by answering yes or no. After checking the prescription and validating the medicine name, the dialog manager again searches for the necessary information on the ontology. This time the retrieved information is the value of the uttered property, such as dosage, frequency, targets, administration route or adverse reactions, for the selected medicine with the posology indicated on the prescription. This information is then sent to the generation module.

If the goal was set as PERMISSION, the dialog manager will immediately check if the uttered medicine name is present on the medical prescription. According to its findings, it will give an appropriate positive or negative answer. If positive, it will also give information about the prescribed frequency for that medication. If negative, the system considers strange the fact that the user asked if he could take a medicine that is not prescribed. Patients tend to ask to take unprescribed medication when they fell unwell. Therefore, the dialog manager identifies a secondary intention behind the user's question and initiates a medical inquiry. The medical inquiry was designed on purpose for HF patients. The objective is to assess if the user has one of the symptoms that characterizes worsen of HF. Some questions are made by the system such as "*Tem pés e pernas inchados*?" (Do you have swollen feet or legs?) or "*Tem falta de ar*?" (Do you have shortness of breath?). The user must answer "*sim*" (yes) or "*não*" (no) accordingly. If the patient answers "*não*" (no) to all the questions, PMA will ask him "*Porque não se sente bem então*?" (Why are you not feeling well then?) so that the user can freely describe its symptoms.

```
**Medical report   in Mon May 20 15:20:24 WEST 2013 **
falta de ar     não
dores no peito  não
cansaço intenso não
aumento de peso superior a 2 kg por semana      não
tosse constante não
pés e pernas inchados   não
Observations - O doente diz: Dói-me muito a cabeça.
```

Figure 4.3: Example of a medical report generated by PMA.

```
**Medical report     Date: Fri May 24 10:51:46 WEST 2013 **
falta de ar     sim
dores no peito      não
cansaço intenso     não
aumento de peso superior a 2 kg por semana      sim
tosse constante     não
pés e pernas inchados       não
Observations: Atenção! Sintomas urgentes
```

Figure 4.4: Example of a medical report generated by PMA for a patient with two of the critical HF symptoms.

If he answered at least two times "*sim*" (yes), the system will tell the user to contact his doctor. Either way, the dialog manager will generate a report with the date, the patient answers each medical inquiry question and an observation field. The observation field can contain the user description of symptoms or the classification of the symptoms as urgent. An example of this medical report which would be sent by email or by message to the user's healthcare staff can be seen in Figure 4.3 and Figure 4.4.

4.2.3 Natural Language Generator

According to the selected keywords and the goal defined by the dialog manager, the generation module will build appropriate sentences to answer the user. The final objective is not only to output informative text but also to produce direct and suitable statements to clarify the user without causing any ambiguity. A group of short and simple sentences will then be preferred to fewer but large and complex phrases.

Natural language generation in PMA is language and application-specific. It exploits grammar rules and lexicons that are specific to healthcare and medical domains. According to the literature, hybrid language generation systems seemed to be more successful and flexible (Galley, Fosler-Lussier, & Potamianos, 2001). When acceptable performance can be achieved without using complex linguist approaches, and only a few types of sentences are being generated, canned text and templates should be sufficient for realization (Reiter & Dale, Building Applied Natural Language Generation Systems, 1997). More complex approaches should make use of MTT, phrase or feature based methodologies. As consequence, PMA tried to implement a hybrid and pratical approach to NLG: simple and typical phrases are generated by canned-text and template-based approaches, while sentences requiring more interaction, initiative and complexity use a combination of phrase and feature-based methodologies.

Canned-text was implemented when fixed instructions or direct answers were required. For example, when an error occurs, a statement is randomly selected from a list of fixed sentences that ask the user to repeat, such as "*Não percebi o que disse. Pode repetir por favor?*" (I did not understand what you said. Can you repeat please?).

The template based approach was used for fixed sentences containing 'gaps' for the information that can differ but takes always the same place on the phrase. This was used, for instance, with the greetings that depend on the time of the day or while listing the prescribed medicines: "*O seu medico receitou <número de medicamentos> medicamentos: <nome medicamento1> <nome medicamento2> ....*" (Your doctor prescribed <number of medicines> medicines: <medicine name1> <medicine name2> …).

Sentence realization includes linguistic and structure realization, where syntactic and morphological rules must be applied to build sentences. The sentences that were generated using phrase and feature based methods are usually the most complex ones, containing the answers that include information from the ontology. Each sentence is specified according to a certain phrase pattern, being composed by the following phrase elements: verb, subject, object, noun modifier, post-modifier and punctuation. Only the elements verb and punctuation are defined as compulsory components of the phrase. Furthermore, generation is dictated by the characteristics of the sentence desired, according to a feature-based methodology. A unique set of features, such as negative/affirmative and singular/plural, must be defined to allow the sentence to be successfully generated.

For getting the appropriate word order, as well as gender and number conjugations, the generator must make use of a lexicon. The XML lexicon contains the necessary information of every word that may be used. Each word element is then characterized by word features like category, number, gender, conjugations, etc… The generator analyses this information and selects the suitable word for the phrase element that it is building, according to the lexicon and the keywords that were recognized. For example, this module automatically chooses the correct word form for a selected noun as well as the determinant article that suits it, in accordance with gender and number. If the noun has an adjective that characterizes it, it will be integrated in conformity. For a verb, the appropriate conjugation is also chosen. Similarly, the object is adapted to the verb used, only appearing if the verb is transitive.

Summarizing module division, the activity diagram from Figure 4.5 shows the flow of tasks in the PMA system.

4.3. Stages of development

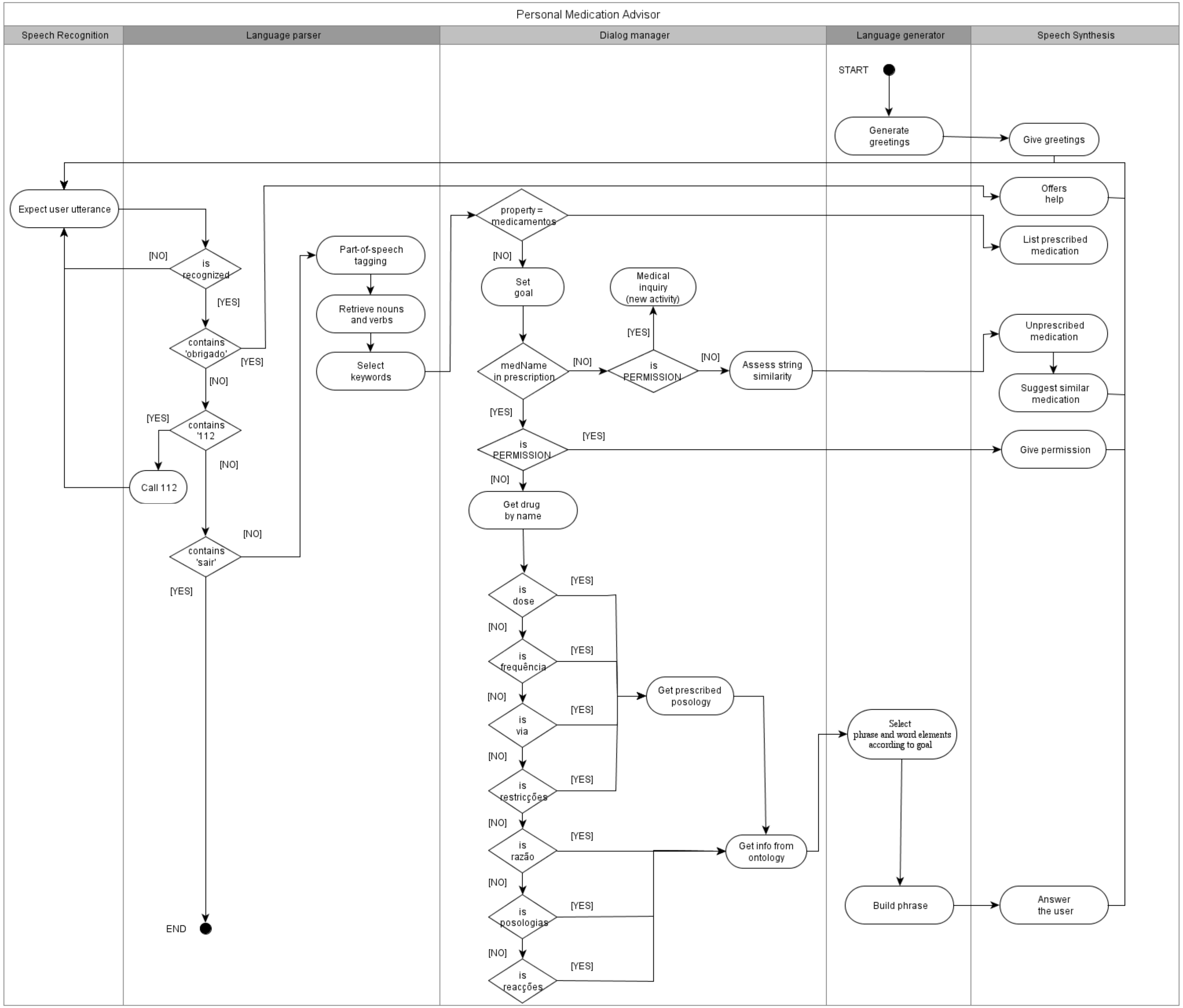
Along the five months of project, the PMA system had three main stages. Each of these phases is characterized by a different system prototype, using different tools and features but with similar architecture.

4.3.1 The first prototype

The initial prototype was a stepping stone into exploring available tools, getting to know better the development platforms and making a preliminary design of system modules. This first version of PMA was English based, using *CMU Sphinx* for speech recognition and *FreeTTS* for speech synthesis. These two modules worked very well when integrated with the remaining system. Speech interaction in English was successful since *FreeTTS* presented very good synthesis in English and *Sphinx* allowed a sufficient recognition based on domain specific Java speech grammar format (JSGF) grammars.

This first prototype only had the basic features allowing information retrieval on posology, adverse reactions, targets, frequency and administration route. This preliminary prototype could only perform canned text generation, this is, fixed strings were outputted according to the goal defined. The language parser used *JTextPro* for tokenization, phrase chunking and part-of-speech tagging. The linguistic resources included an English dictionary bundled with *Sphinx*, which was slightly extended by adding the information on some domain specific words that were missing. This prototype was merely experimental as the ontology was in Portuguese and the system only allowed a rigid and very limited Q&A dialog.

Figure 4.5: Simplified UML activity diagram of PMA.



4.3.2 The second prototype

On a second stage, the system was developed in Java, also for PC, but this time using Brazilian Portuguese and including more features. This version of PMA was the one that took longer to develop as it tried to come closer to a mixed-initiative dialogue system, expecting some active interaction from both human user and intelligent assistant.

This version already had all the features comprised along the previous sections. Besides the basic Q&A information retrieval on medication dosage, frequency, administration route, adverse reactions, restrictions and targets, there is also the possibility of checking prescribed medication and know what medicines can be taken. The medical inquiry and consequent generation of medical report were firstly included in this prototype.

For Portuguese, *TreeTagger* was used by the language parser to do the part-of-speech tagging. Both language parsing and generation were built while thinking with the grammar rules of European Portuguese, which would be the language of the final prototype.

Speech synthesis was performed with *FurbSpeech* while *CMU Sphinx,* with modified grammar files and lexicons, was integrated for recognition. Speech interaction in Brazilian Portuguese was not as successful as in English, especial due to the recognition module. As explained on the *Speech tools* section, *FurbSpeech* synthesis uses MBROLA voices that work very well in Brazilian Portuguese but are poorly acceptable in European Portuguese. Furthermore, recognition in Portuguese was insufficient due to the inexistence of freely available and trained acoustic models for this language. As no other solutions were found for this platform, written input was temporarily enabled to facilitate the development and testing of the other modules. A few examples of interactions with this version of PMA are present in *Appendix A.*

4.3.3 The final system

The final PMA is an adaptation of the second prototype to the Android platform. Using Android speech packages and *iSpeech* android SDK, speech interaction was now successfully available in European Portuguese with very good synthesis and good recognition. The incorporation of PMA in the Android platform required several changes from the second version, although most of the routines and features were maintained. It is also important to note that, due to the nature of the speech recognition technology used, domain-specific JSGF grammars could not be used for recognition in this prototype. With this, a disadvantage of this system, compared with the previous prototypes, was the limitation of Google Voice which does not allow to add new vocabulary to be recognized.

This version has exactly the same features as the previous one. However, the linguistic processing had to be reformulated since there are no Portuguese taggers available for Android platforms. A simple lexicon based tagger was built from scratch and, although it presents limitations, it was sufficient for the purpose of this language parser.

The interface of this application tried to be as simple as possible. Some screenshots of the application are shown in Figure 4.6. There is a screen to initialize the assistant and, after clicking the *Start* button, the PMA activity, where the dialog will occur, opens. The user has to press a button every time he wants to speak and a Google Voice popup window will briefly appear for speech recognition. As the dialog proceeds, everything that was said by the user or by the system is shown on the screen within a dialog list that can be scrolled up and down. A button for stopping the assistant when it is talking was also added.

4.4. Summary

Along this chapter, the main features, structure and characteristics of the whole PMA were described. The system can give information on medication dosage, frequency, administration route, targets, restrictions and adverse reactions, while having the ability to distinguish and notify about the medication that was prescribed by the patient's doctor. It can also perform a medical inquiry. PMA arrangement is divided in five modules, being the recognizer, the language parser, the dialog manager, the language generator and the synthesizer. Each one has its individual tasks but all needed for the system. The project had three phases of development. The first was only an experimental approach; the second already included all the features; the final one, on the Android platform, was a modification of the second integrated with more efficient European Portuguese speech tools. A successful system was then achieved.

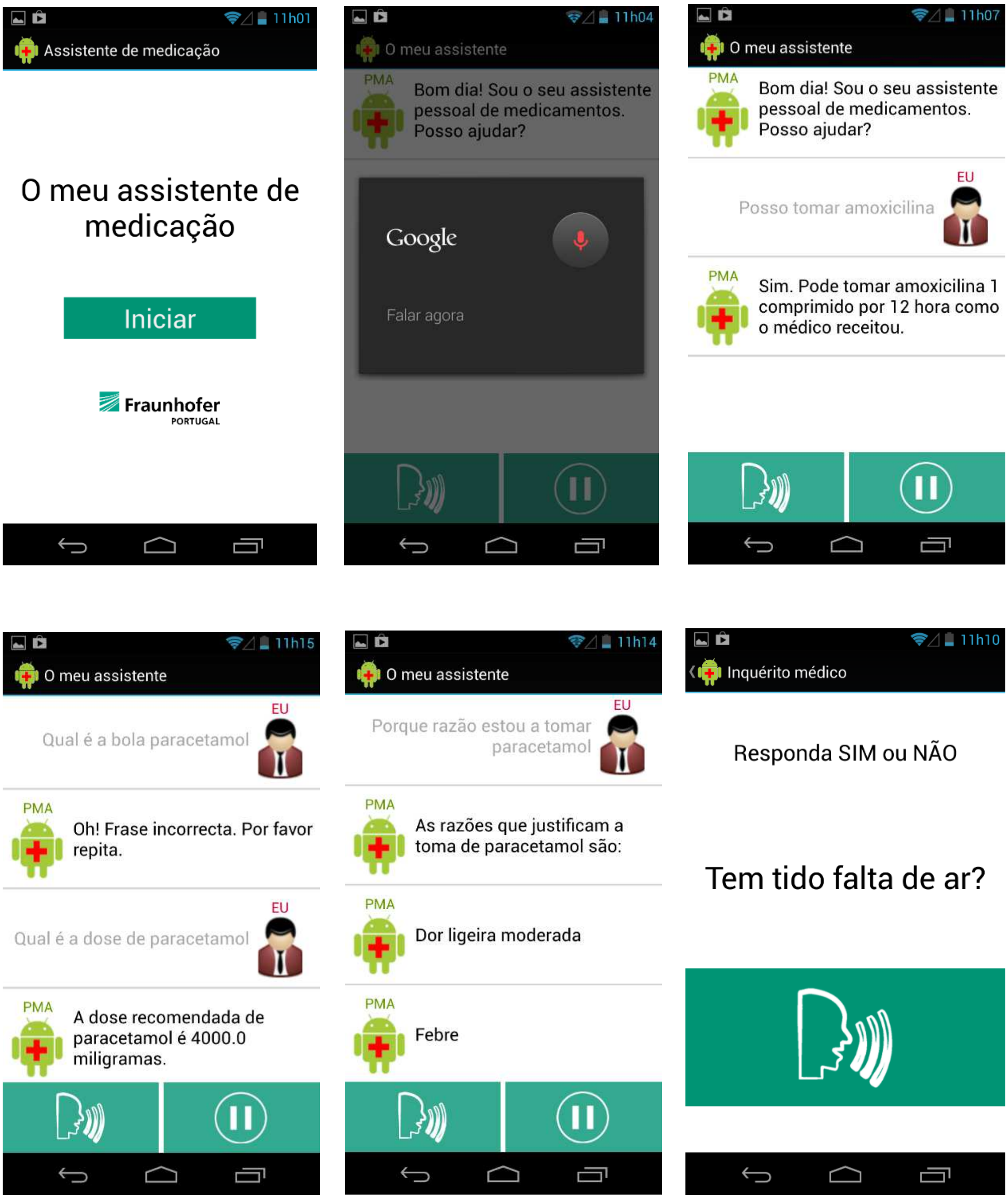


Figure 4.6: Screenshots with examples of interaction with PMA Android application, featuring a) welcome screen, b) Google Voice popup window, c) d) e) examples of dialogue, f) one step of the medical inquiry.

Chapter 5

Evaluation and Discussion

A simple conceptual evaluation was made to assess PMA. This chapter, first, presents the evaluation rationale, then the methodologies used and the results obtained. The limitations of the developed system, along with some thoughts on the overall project are detailed in the second part of this chapter.

5.1. Evaluation

The rationale of evaluating systems like PMA relies in collecting real data, analysing it and then improving system performance from the flaws identified. Nevertheless, evaluating a conversational system, especially understanding and generation modules, is still a challenging research problem (Baptist, 2000). The problem of evaluating conversational assistants may lie in the limited comparability of results and the impossibility of a full formal evaluation due to the heavy dependence on the context where the dialog happens (Artstein, Gandhe, Gerten, Leuski, & Traum, 2009). Although still far from a satisfactory assessment of a conversational system, there are several quantitative or qualitative perspectives of analysis that can be made so at least a rough judgment of system performance is available.

Quantitative evaluation could involve measuring simple dimensions of the system like speed and memory requirements or other basic parameters like dialogue length or task success rate. Some performance measures for natural language dialogue systems can be found on the literature, either for generation, (Gatt & Port, 2009) recognition modules (Smith, 1997) or the overall system. (Kamm, Walker, & Litman, 2000) (Danieli & Gerbino, 1995). Temporal measurements, such as the time taken to fulfil a task with the system, may not be good predictors of user satisfaction (Walker, Boland, & Kamm, 1999). A qualitative evaluation seems to be much more difficult to achieve (Baptist, 2000). One can measure the quality of the generated strings, system flexibility or its eases of use. For NLU and NLG modules, the scalability, tailorability and adaptability can also be roughly estimated. A subjective evaluation can also appraise system's ability, for driving the user to his goal, and its robustness due the implemented error recovery strategies.

Even so, the best indication of a system's quality is the collective experience of those who use it (Baptist, 2000). Following this idea, testing the usability of PMA with a few senior patients would be the best panorama for assessing system feasibility. But as the system is still in an early phase of development and tests with older people require special procedures, it was decided that usability tests for PMA would not include senior patients for the time being. System usability can be mainly assessed either by distributing a questionnaire to the users to record their subjective assessment of using the dialogue system or by studying the resulting dialogue. The first alternative was here preferred.

5.1.1 Methodology

Eleven volunteers, with ages between 22 and 30, participated on the usability study of the PMA application, tested on a Samsung Galaxy Nexus GT-I9250 with Android 4.0. The usability test consisted in individually asking the user to accomplish two simulated tasks and then filling a questionnaire, available in

Appendix B, to register his subjective perception of his interaction with PMA. The objective was not to assess recognition or synthesis modules, but rather the design, task success and user satisfaction. Is spoken interaction useful and enjoyable? Does the flow of the application match expectations? Is one able to find the desired information? Does the interface make sense to the users?

Before starting the usability test, a brief contextualization of PMA was made to each user. It was asked to participants to simulate a heart failure medication scenario. The user would play a patient with medication prescribed for heart failure. He was given a list of four medicines that he would have at home (Eplerenone, Bisoprolol, Amoxiciline and Paracetamol), not necessarily medication that was prescribed. A list of the possible utterances to interact with PMA was also given to the user. It was also said that one should only take a medicine that was prescribed by the doctor. The user-system interaction was recorded with a camera during the tasks. With this, the first task was stated as:

"You have a fever. What should you do? Can you take one of the medicines you have at home? If yes, which one?"

To complete the task, the user should answer the questions above. This first task would require asking a minimum of two questions to PMA. The intention was that the user would check with PMA what was prescribed and then ask about the reason for taking each of those medicines. He should conclude that he should take Paracetamol because it was prescribed and one of its targets is fever.

For the second task, it was said to the user that:

"Your legs are swollen and you feel very tired. Someone said that the unprescribed medicine that you have at home can reduce swelling. What should you do? Should you take that medicine?"

This task would require asking a minimum of two questions to PMA as well as answering some yes/no questions that the system asks. The objective of this task was for the user to test what the system does when he asks if he can take a medicine that is not prescribed. With this, the feature of the medical inquiry is explored and the user must have concluded that he should contact a doctor immediately due to his symptoms, as PMA advises.

After completing both tasks, it was given an opportunity to freely explore and use PMA. After that, the questionnaire, with 16 statements divided in two sections, was given. Each statement was evaluated by the participants with a score between 1 (strongly agree) and 4 (strongly disagree). The first twelve questions are related to the satisfaction of the user. Questions 1, 2, 3, 7 and 12 evaluate usability and likability, while 4 and 5 evaluate coherence and appropriateness. Question 6 evaluates naturalness, as 8, 9, 10 and 11 the quality of the information given by the system. Questions from 13 to 16 are focused on a senior user perspective. To conclude the test, a time for commentaries and suggestions was given.

The first part of the questionnaire was partially based on the System Usability Scale questionnaire (SUS) (Brooke, 1986). Furthermore, the usability test was planned in accordance with some of the Fraunhofer guidelines for a smart companion.

5.1.2 Observations

During the testing, some difficulties with recognition were observed for the majority of the participants, especially related to medicine names such as "*eplerenona*". Nevertheless, all the users ended up completing both tasks, some of them more easily and faster than others. It was also interesting to note that one of the participants had some speech disfluencies and PMA was nonetheless able to understand what he said.

For both tasks, there were several ways of getting the required information, even if some of them were straighter and faster than the others. The first task registered an average of 5 questions per user necessary to achieve the goal. It was observed that, in this task, participants spent a lot of time asking other questions unrelated to the objective. The second task presented an average of 4 questions and users were more comfortable using the system. A table with more information on test data can be seen on *Appendix A.II/III*. The average scores obtained for each point on the questionnaire are showed in Figure 5.1.

Questions 8 and 13 had the most consensual answers. All the participants felt that PMA was easy to use and its interface was pleasant and simple. Overall, quality of information, coherence and naturalness were classified as good. There was not much consensus on the usability of PMA for older people, especially due to the disparity of results from questions 14 and 16. It was also registered that the most used question during usability tests was: "*Posso tomar* <*nome de medicamento*>?" (Can I take <*medicine name*>?).

Overall, user engagement with PMA seemed to be successful. However, the majority of the participants said that they felt that PMA was not very intuitive at the beginning. Only after some time of interaction with the system, they got to understand how it worked and then it got easy to use. Some of them even said that PMA should come initially with some embedded instructions. It was also observed that some participants wanted to forcefully talk with the system using non contemplated utterances because they were more natural for them. Moreover, they also considered that asking directly by the symptoms would be better. For example, they wanted to say 'I have a fever' or 'I have a headache' and PMA would answer with the medicine that is prescribed for that symptom.

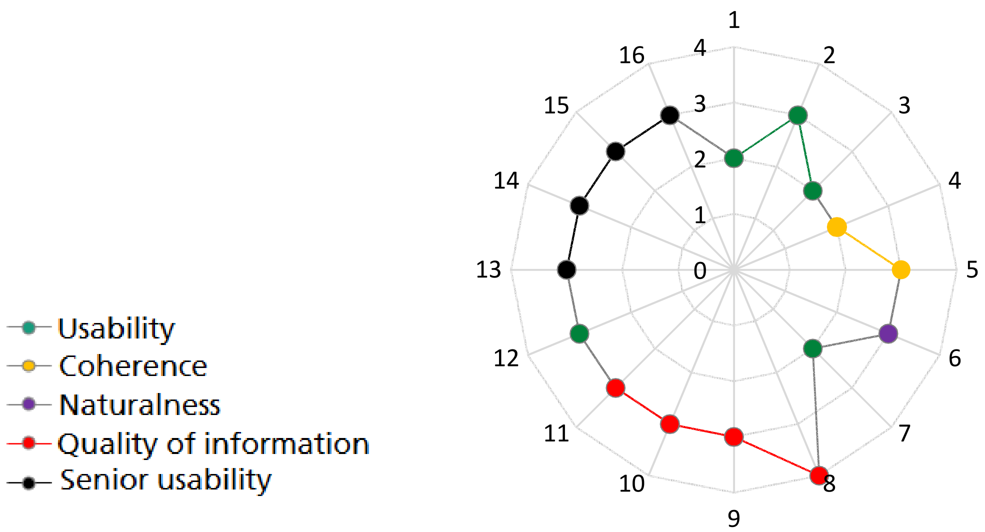


Figure 5.1: Average score (radius) given by the participants of the usability test to each of the 16 questionnaire's statements (perimeter) with the respective attribute that each question assesses.

Those who had working experience with senior people seemed to think that older patients would not learn how to use PMA quickly, doubting that they would use it for daily-life. In the end they said that some modifications should be introduced to further adapt PMA for an older audience.

It was also suggested that it would be interesting to integrate PMA with visual information or even a barcode reader for medicine boxes. As some senior patients are not able to read, they may not know the name of their medication, written on the medicine packages or their prescriptions. Some participants said that some type of information as medication-medication interactions would also make the system more helpful.

The usability test was performed with a small sample size, what is not particularly accurate for this type of questionnaire. However, it was enough to have a preliminary idea of how PMA can improve its reliability, helpfulness and usability. The conclusions derived from the usability testing will allow refining PMA and should be seen as future work.

5.2. General discussion

5.2.1 System Limitations

The developed system presents some limitations that are important to be aware of. Overall, PMA lacks in flexibility. The system should be able to understand a wider variety of questions and user utterances as well as have a more natural flow of conversation. Android implementation brought PMA closer to a mixed-initiative conversational system. Even so, the system should still have more initiative to improve user engagement.

In what relates to speech tools, a limitation is the fact that PMA needs to be connected to a network service provider so that speech tools can be used. Unfortunately, this cannot be avoided. In fact, this can be an advantage since web services can make the system faster.

For successful recognition, the user must speak loud, clearly and not too fast. In addition, he must only speak when the assistant is silent, otherwise it will not work well. Speech recognition does not need to be perfect in order to be effective (Ferguson, Personal Health Management Assistant, 2007). Natural human-human conversations also have misunderstandings. However, in the current PMA it still occurs too often and affects system performance. The major recognition problem occurs with medicine names. More work could be done in the direction of correcting recognition errors through the introduction of a speech checker to suggest possible replacements for unrecognized words or to complete questions. Currently, PMA only does so when there is a recognition error on a medicine name.

PMA is also slow to start because it has to load the whole ontology during application launching. This would be solved by using web services to get ontology information. Other limitation lies in the fact that the information contained in the ontology is extracted from a therapeutic compendium book and thus it is not specific for that patient (not personalized) being sometimes not so suitable for that user.

5.2.2 Other considerations

A reasonable question can be raised for discussion: would PMA be used by older people? Some barriers are present in healthcare systems whose target is elderly population. Since older patients do not have easy access, technical literacy or familiarity with technology in general, the adherence to a computerized-assistant could be poor. Senior patients may view the technology as intrusive and not suitable with their lifestyle, offering resistance to the introduction of new equipment or new habits. They are also overly attached to their daily routines and it would take a lot of time to feel comfortable with the developed technology. Furthermore, as it was concluded from the usability testing, initial training could be required.

At the NLG level, the complexity of PMA has not yet arrived to a stage where stochastic models are needed for generation. Simultaneously, generation is made phrase by phrase, still far away from the desired multi-sentence generation. A spell checker for the generated sentences could also be implemented to recognize possible structure mistakes or misconceptions from users' queries.

It is important that PMA remains a medical information-provider rather than a decision-maker. A conversational system should never be a physician's substitute. 'Technology should be developed to complement and augment human decision-making' (Ferguson, Personal Health Management Assistant, 2007). Furthermore, it is important to maintain a good granularity for the information given by the intelligent assistant. The type and amount of information that is appropriated to give may not be the same for every patient. For example, the long list of the adverse effects presented on prescribed medicines can pointlessly stress patients, even when most of these events are rare or unworthy of attention. Once again, further personalization of PMA could avoid causing these unnecessary worries or anxiety. The information that is being given to the patient should be adapted and given according to some priorities and to the social context of the patient, so that problems of excess of information can be avoided. This adjustment can also make the assistant more robust (Jones, Cawsey, & Bentala, 1999).

5.3. Summary

A semi-formal evaluation of PMA was performed to assess usability and detect its drawbacks so that improvements can be made in the future. Participants were asked to complete two tasks with the developed android application and then fill a usability questionnaire. It was concluded that users felt that the system was easy to use, simple and had a good interface. Participants also gave several suggestions that could help improving PMA.

The system still presents limitations such as lack of flexibility, need a wireless connection to the network, or frequent recognition mistakes. System helpfulness for senior users was also discussed.

Chapter 6

Conclusions

After five months of project development, the Personal Medication Assistant culminated in an accessible android application that interacts with the user through voice to give information about medication. In this last chapter, a balance of the overall project is made: from the evaluation of the initial objectives to the main contributions and the challenges found. Possible futures for PMA refinement are also discussed.

6.1. Final Balance

The major objective of this project was to come nearer to a conversational assistant: the Personal Medication Advisor, a system capable of real-time interaction with a user to help him manage his medication through spoken natural language in European Portuguese. Making use of speech tools, ontologies and linguistic resources as grammars and lexicons, the final system resulted of iterations during three stages of development. The last prototype took the form of an Android application where the user can successfully dialog with the system using voice. PMA structure is composed by five modules: a recognizer, a language parser, a dialog manager, a language generator and a synthesizer. Both recognizer and synthesizer modules are external using, in the final PMA version, *Google Voice* for speech recognition and *iSpeech* for speech synthesis. Natural language understanding was achieved by the language parser using phrase chunking, part-of-speech tagging and part of keyword spotting. Single-sentence natural language generation was achieved through hybrid methodologies that use simultaneously canned-text, template-based and feature-based approaches.

The PMA application was validated through a usability test. This evaluation wanted to assess system's feasibility, performance and interface, through participants' opinions after completing two tasks and filling a suitable questionnaire. Task objectives were successfully achieved. It was concluded that users felt that the system was simple and had a good interface.

If this project moves forward, further adaptations for senior users must be made to make PMA more helpful and accessible for them.

Initially, this work was meant to be focused on the development of a language generator module capable of producing sentences to be synthesized by existent TTS systems. This objective was accomplished and expanded to the incorporation of a language understanding and dialog management module with existing ASR systems. This was because functional language parsing and managing were needed for achieving any type of conversational assistance.

With this, all the functionalities planned in the beginning for PMA were achieved. The objective of having successful integration between NLU/NLG modules and speech tools was also accomplished. As proposed, PMA generates phrases that are coherent, short and easy to understand.

Strengths & weakness

PMA's good points are inherent in its motivation, its architecture that aimed to enhance modularity, its simplicity and its pleasant interface. The uniqueness of PMA lies on the use of ontologies as source of medical information.

On the other hand, the system still lacks in flexibility, initiative and naturalness, compared to users' expectations of a conversational assistant. The system should require no special training for the patients that are going to use it. However, the usability test showed that for senior users some features need to be slightly modified to facilitate their first interaction with PMA.

Contributions

The system was envisioned within a patient centred approach to personal healthcare, namely Heart Failure care, so that home-monitoring and treatment quality and efficacy can be enhanced. The specific evidence-based information given by PMA can help patients making positive behaviour changes. For example, the available information will allow users to know possible negative effects of their medication, not being caught completely unaware. The simplicity and interactivity of the system will trigger user engagement and hopefully improve medication adherence. User satisfaction with PMA will raise their self-consciousness of home healthcare and work together with physicians to modify disease outcome.

Challenges encountered

During the five months of project, work evolved dynamically as the system was being developed. Nevertheless, major challenges appeared during the different stages of PMA. The major obstacle for this project was associated with the lack of freely available speech tools for European Portuguese. Several tools were assessed and only the final version was able to have successful recognition in this language. Additionally, compared to other languages, European Portuguese is still very immature in what is related to natural language research, especially NLG. Herewith, generation presented a great challenge.

Android implementation of PMA also required an extra effort because it was necessary to remodel part of what had already been developed on previous stages. This was also a new challenging platform of development, since it was the first contact with it.

6.2. Future work

As PMA has limitations, further system development can be made. Due to the complex nature of conversational assistants, there are always new features to add or improvements to make. This is especially true for natural language understanding and generation. Granularity of utterance's interpretation as well as of the phrase elements that will be part of generated sentences can be iteratively diminished and refined. This will gradually result in a more complex and enhanced system. Step by step, it should also smooth some of the system limitations, such as the lack of flexibility, the lack of initiative or of naturalness. One of these steps could be the implementation of statistical and/or stochastic methodologies. Others include giving the system the ability to learn common user utterances or underlying intentions, building more complete approaches for language generation, or introducing an automated vocabulary growth feature. For example, the ability to learn new words is important for interactive systems and will increase their flexibility.

PMA needs to be more tailored to each patient needs. A major improvement at this level would be the incorporation of user's medical history and past medical records. This could be complemented with a variety of useful data about the patient. Basic information such as name, age, sex, educational level, local background and physiological data (height, weight, blood pressure, ….) could be easily provided. Recognition and synthesis of user's name would also make the interaction more natural and personalized. With a great increase of complexity, additional intelligent modules could be useful to detect speaker identity, emotional state or even degree of motivation. The system could be programmed to use this information and adjust its decisions and dialogue accordingly. This attempt to personalize the assistance would be very advantageous, although it would present a bigger challenge.

Further adaption of PMA to senior users is still necessary. From modification of the type of utterances allowed to the integration of visual information on the medicine appearance, more work can be done to increase acceptance and convenience of PMA for older patients. Incorporation with already existent medication reminders or alerts would also boost system helpfulness. Moreover, as stated in the introduction, a second and advanced approach to the Personal Medication Assistant will have to include contraindications, cautions and drug to drug interactions.

References

Aguiar, B. L. (2012). *Information Extraction From Medication Leaflets.* Master Thesis, FEUP, Porto.

Allen, J. F., Byron, D. K., Dzikovska, M., Ferguson, G., Galescu, L., & Stent, A. (2001). Toward conversational human-computer interaction. *AI Magazine, 22*(4), 27-38.

Allen, J., Ferguson, G., & Stent, A. (2001). An Architecture For More Realistic Conversational Systems. *6th international conference on Intelligent user interfaces* (pp. 1-8). New Mexico: ACM New York.

Allnatt, R., Engelhardt, J. B., Gao, J., & Mariano, A. (2001). An Evaluation of the Functionality and Acceptability of the Voice Prescription Label. *Journal of Visual Impairment and Blindness, 95*(11), 702-706.

Axelrod, S. (2000). Natural language generation in the IBM Flight Information System. *ANLP/NAACL, Workshop on Conversational Systems*, *3*, pp. 21-26. Seattle.

Bangalore, S., & Rambow, O. (2000). Corpus-based lexical choice in natural language generation. *38th Annual Meeting of the Association for Computational Linguistics*, (pp. 464-471). Hong Kong.

Bangalore, S., & Rambow, O. (2000). Exploiting a probabilistic hierarchical model for generation. *18th International Conference on Computational*, (pp. 42-48). Saarbrücken, Germany.

Baptist, L. M. (2000). *Genesis II: a language generation module for conversational systems.* Master Thesis, MIT, Department of Electrical Engineering and Computer Science, B.A.

Baptista, J., Mamede, N., & Gomes, F. (2010). Auxiliary Verbs and Verbal Chains in European Portuguese. *9th International Conference of Computational Processing of the Portuguese Language*, (pp. 110-119). Porto Alegre.

Benzeghiba, M., Mori, R. D., iDeroo, O., Dupont, S., Erbes, T., Jouvet, D., . . . Wellekens, C. (2007). Automatic speech recognition and speech variability: A review. *Speech Communication, 49*(10-11), 763–786.

Brooke, J. (1986). *SUS – A quick and dirty usability scale.* United Kingdom: Redhatch Consulting.

Cardoso, P., Flores, L., Langlois, T., & Neto, J. (2002). Meteo: A Telephone-Based Portuguese Conversation System in Weather Domain. *Third International Conference of Advances in Natural Language Processing, PorTAL*, (pp. 175-178). Faro.

Carolis, B. D., Rosis, F. d., Grass, F., Rossiello, A., Berry, D. C., & Gillie, T. (1996). Generating recipient-centered explanations about drug prescriptions. *Artificial lntelligence in Medicine, 8*, 123-145.

Carvalho, P., Oliveira, L. C., Viana, M. C., & Trancoso, I. (1998). *Concatenative speech synthesis for European Portuguese.* Lisbon: INESC/ICT.

Ceia, F., Fonseca, C., Mota, T., Morais, H., Matias, F., Costa, C., & Gouveia-Oliveira, A. (2004). Epidemiology of heart failure in mainland Portugal: new data from the EPICA study. *Revista Portuguesa de Cardiologia, 23*(3), 15-22.

Chambers, N., & Allen, J. (2004). Stochastic Language Generation in a Dialogue System: Toward a Domain Independent Generator. *5th SIGdial Workshop on Discourse and Dialogue.* Boston.

Cole, R., Mariani, J., Uszkoreit, H., Varile, G. B., Zaenen, A., & Zampolli, A. (1996). *Survey of the State of the Art in Human Language.* Cambridge University Press.

Corston-Oliver, S. (2000). Using Decision Trees to Select the Grammatical Relation of a Noun Phrase. *1st SIGDial workshop on discourse and dialogue*, *10*, pp. 66-73. Hong Kong.

Dale, R. (1995). An Introduction to Natural Language Generation. *European Summer School in Logic, Language and Information.* Barcelona.

Danieli, M., & Gerbino, E. (1995). Metrics for evaluating dialogue strategies in a spoken language system. *AAAI Spring Symposium on Empirical Methods in Discourse Interpretation and Generation*, *16.*

Davis, H. K., Biddulph, R., & Balashek, S. (1952). Automatic Recognition of Spoken Digits. *Journal of the Acoustical Society of America, 24*(6), 627-642.

Dudley, H. (1939). The Vocoder. *Bell Labs Record, 17*, 122-126.

Ferguson, G. (2007). *Personal Health Management Assistant.* PhD Thesis, University of Rochester.

Ferguson, G., Allen, J., Blaylock, N., Byron, D., Chambers, N., Dzikovska, M., . . . Swift, M. (2006). Chester: towards a personal medication advisor. *Journal of biomedical informatics, 39*(5), 500-513.

Ferguson, G., Quinn, J., Horwitz, C., Swift, M., Allen, J., & Galescu, L. (2010). Towards a Personal Health Management Assistant. *Journal of biomedical informatics, 43*(5 Suppl), S13-16.

Galescu, L., Allen, J., Ferguson, G., Swift, M., & Quinn, J. (2009). Speech Recognition in a Dialog System for Patient Health Monitoring. *IEEE International Conference on Bioinformatics and Biomedicine - Workshop on NLP Approaches for Unmet Information Needs in Health Care.* Washington D.C.

Galley, M., Fosler-Lussier, E., & Potamianos, A. (2001). Hybrid Natural Language Generation for Spoken Dialogue Systems. *European Speech Processing Conference*, (pp. 1735–1738). Aalborg.

Gatt, A., & Port, F. (2009). Text content and task performance in the evaluation of a Natural Language System. *International Conference on Recent Advances in Natural Language Processing.*

Giles, T. D. (2004). New perspectives in the prevention and treatment of chronic heart failure with reduced systolic function. *American Journal of Hypertension, 17*(1S), 249A.

Goldberg, E., Driedger, N., & Kittredge, R. (1994). Using natural-language processing to produce weather forecasts. *IEEE Expert, 9*(2), 45-53.

Grancelli H, V. S. (2003). Randomized trial of telephone intervention in chronic heart failure (DIAL): study design and preliminary observations. *Journal of Cardiac Failure, 9*(3), 172-179.

Haux, R. (2006). Health information systems – past, present, future. *International Journal of Medical Informatics, 75*(3-4), 268–281.

Heart Failure Association of the ESC. (2010). EURObservational Research Programme: The Heart Failure Pilot Survey. *European Journal of Heart Failure, 12*, 1076–1084.

Jelinek, F. (1997). *Statistical Methods for Speech Recognition.* MIT Press.

Jimison, H., Gorman, P., Woods, S., Nygren, P., Walker, M., Norris, S., & Hersh, W. (2008). *Barriers and drivers of health information technology use for the elderly, chronically ill, and underserved.* Rockville: Agency for Healthcare Research and Quality.

Jones, R., Cawsey, A., & Bentala, D. S. (1999). Patient information systems that tailor to the individual. *Patient Education and Counseling, 36*, 171–180.

Kamm, C., Walker, M., & Litman, D. (2000). *Evaluating Spoken Language Systems.* AT&T Labs Research.

Krumholz, H. M., Chen, Y. T., Wang, Y., Vaccarino, V., Radford, M. J., & Horwitz, R. I. (2000). Predictors of readmission among elderly survivors of admission with heart failure. *American Heart Journal, 139*(1), 72-77.

Lemmetty, S. (1999). *Review of Speech Synthesis Technology.* Master Thesis, Helsinki University of Technology, Helsinki.

Mavridis, N. (2007). *Grounded Situation Models for Situated Conversational Assistants.* PhD thesis, Massachusetts Institute of Technology, Massachusetts.

Meinedo, H., Caseiro, D., Neto, J. P., & Trancoso, I. (2003). AUDIMUS.MEDIA: A Broadcast News Speech Recognition System for the European Portuguese Language. *6th International Workshop of Computational Processing of the Portuguese Language.* Faro.

Mellish, C., & Evans, R. (2004). Implementation architectures for natural language generation. *Natural Language Engineering, 10*(3-4), 261 - 282.

Mettler, M., & Kemper, D. W. (2003). Information therapy: health education one person at a time. *Health promotion practice, 4*(3), 214-217.

Michalsen, A., König, G., & Thimme, W. (1998). Preventable causative factors leading to hospital admission with decompensated heart failure. *Heart, 80*, 437–441.

Moore, J. D., & Paris, C. L. (1993). Planning text for advisory dialogues: Capturing intentional and rhetorical information. *Computational Linguistics, 19*, 651-694.

Naylor, M. D., Brooten, D. A., Campbell, R. L., Maislin, G., McCauley, K. M., & Schwartz, J. S. (2004). Transitional care of older adults hospitalized with heart failure: A randomized control trial. *Journal of American Geriatric Society, 52*, 675-684.

Neto, J., & Cassaca, R. (2004). A Robust Input Interface in the scope of the Project Interactive Home of the Future. *ROBUST 04.* Norwich.

Nieuwenhuis, M. W., Jaarsma, T., van Veldhuisen, D. J., & van der Wal, M. L. (2012). Self-reported versus 'true' adherence in heart failure patients: a study using the Medication Event Monitoring System. *Netherlands Heart Journal, 20*(7-8), 313–319.

Oh, A. H., & Rudnicky, A. I. (2002). Stochastic natural language generation for spoken dialog systems. *Computer Speech and Language, 16*, 387-407.

Olson, F. H., & Belar, H. (1956). Phonetic Typewriter. *Journal of the Acoustical Society of America, 28*(6), 1072-1081.

Phan, X.-H. (2006). Retrieved March 2013, from JTextPro: A Java-based Text Processing Toolkit: http://jtextpro.sourceforge.net

Raistrick, Phillip C; Raistrick, David; En-Vision America. (2006). Audible Prescription Reading Devices: Solving Issues In Privacy, Safety And Health. *International Conference on Aging, Disability and Independence.* St. Petersburg.

Reiter, E. (1995). NLG vs. templates. *Fifth European Workshop on Natural Language Generation.* The Netherlands.

Schroeder, M. R. (1993). A Brief History of Synthetic Speech. *Speech Communication, 13*(1-2), 231–237.

Seneff, S. (2003). Spoken Computer conversational systems. *JHU Summer Language Workshop.* Massachusetts: CSAIL, MIT.

Silva, J., Branco, A., Castro, S., & Reis, R. (2010). Out-of-the-Box Robust Parsing of Portuguese. *9th International Conference of Computaional Processing of the Portuguese Language*, (pp. 75-85). Porto Alegre.

Simon, S., & Horowitz, J. (2002). Home-based intervention in congestive heart failure: long-term implications on readmission and survival. *Circulation, 105*(24), 2861-2866.

Smedt, K. D., Horacek, H., & Zock, M. (1996). Architectures for Natural Language Generation: Problems and Perspectives. *Trends in natural language generation: an artificial intelligence perspective* (pp. 17-46). Springer-Verlag.

Smith, R. W. (1997). Performance Measures for the Next Generation of Spoken Natural Language Dialog Systems. *Workshop On Interactive Spoken Dialog Systems: Bringing Speech And NLP Together In Real Applications.* Madrid.

Staab, S., & Studer, R. (Eds.). (2009). *Handbook on Ontologies.* Berlin: Springer.

Stewarta, S., Jenkins, A., Buchan, S., McGuire, A., Capewell, S., & McMurray, J. J. (2001). The current cost of heart failure to the National Health Service in the UK. *European Journal of Heart Failure, 4*(3), 361-371.

Studer, R., Benjamins, R., & Fensel, D. (1998). Knowledge engineering: Principles and methods. *Data & Knowledge Engineering, 25*(1-2), 161–198.

Sullivan, R., Behncke, I., & Purushotham, A. (2010). Why do we love medicines so much? *Science and Society, 11*(8), 572–578.

Thakur, B. K., Chettr, B., & Shah, K. B. (2012). Current Trends, Frameworks and Techniques Used in Speech Synthesis – A Survey. *International Journal of Soft Computing and Engineering,, 2*(2), 2231-2307.

Topol, E. J. (2010). Transforming Medicine via Digital Innovation. *Science Translational Medicine, 2*(16).

Wagner, J. C., Rogers, J. E., Baud, R. H., & Scherrer, J.-R. (1999). Natural Language Generation of Surgical Procedures. *International Journal of Medical Informatics, 53*(2-3), 175-192.

Walker, M., Boland, J., & Kamm, C. (1999). The utility of elapsed time as a usability metric for spoken dialogue systems. *IEEE Workshop on Automatic Speech Recognition and Understanding.*

Walker, M., Rambow, O., & Rogati, M. (2001). SPoT: A trainable sentence planner. *Annual Conference of the North American Meeting of the Association for Computational Linguistics* (pp. 1-8). Stroudsburg: Association for Computational Linguistics.

Appendix A

Table A. I: Aetiology of heart failure. Source: (European Society of Cardiology, 2012)

Web Table 3 Aetiology of heart failure

There is no agreed or satisfactory classification for the causes of HF, with much overlap between potential categories

Myocardial disease

1. Coronary artery disease
2. Hypertension[a]
3. Cardiomyopathy[b]
    - a. Familial
        - i. Hypertrophic
        - ii. Dilated
        - iii. Arrhythmogenic right ventricular cardiomyopathy
        - iv. Restrictive
        - v. Left ventricular non-compaction
    - b. Acquired[c]
        - i. Myocarditis (inflammatory cardiomyopathy)
            - Infective
                - Bacterial
                - Spirochaetal
                - Fungal
                - Protozoal
                - Parasitic
                - Rickettsial
                - Viral
            - Immune-mediated
                - Tetanus toxoid, vaccines, serum sickness
                - Drugs
                - Lymphocytic/giant cell myocarditis
                - Sarcoidosis
                - Autoimmune
                - Eosinophilic (Churg–Strauss)
            - Toxic
                - Drugs (e.g. chemotherapy, cocaine)
                - Alcohol
                - Heavy metals (copper, iron, lead)
        - ii. Endocrine/nutritional
            - Phaeochromocytoma
            - Vitamin deficiency (e.g. thiamine)
            - Selenium deficiency
            - Hypophosphataemia
            - Hypocalcaemia
        - iii. Pregnancy
        - iv. Infiltration
            - Amyloidosis
            - Malignancy

Valvular heart disease

- Mitral
- Aortic
- Tricuspid
- Pulmonary

Pericardial disease

- Constrictive pericarditis
- Pericardial effusion

Endocardial disease

- Endomyocardial diseases with hypereosinophilia [hypereosinophilic syndromes (HES)]
- Endomyocardial disease without hypereosinophilia [e.g. endomyocardial fibrosis (EMF)]
- Endocardial fibroelastosis

Congenital heart disease

Arrhythmia

- Tachyarrhythmia
    - Atrial
    - Ventricular
- Bradyarrhythymia
    - Sinus node dysfunction

Conduction disorders

- Atrioventricular block

High output states

- Anaemia
- Sepsis
- Thyrotoxicosis
- Paget's disease
- Arteriovenous fistula

Volume overload

- Renal failure
- Iatrogenic (e.g. post-operative fluid infusion)

AV = atrioventricular; HF = heart failure.
[a]Both peripheral arterial and myocardial factors contribute to the development of heart failure.
[b]Other inherited diseases may have cardiac effects. e.g. Fabry disease.

```
******* Assistente a iniciar. Aguarde ********
PMA disse:  bom dia! Sou o seu assistente pessoal de medicação. Em que posso ajudá-lo?
User diz:  qual é a dosagem de enalapirl
PMA disse:  Não percebi o que disse. Pode repetir?
User diz:   qual é a dosagem de enalaprol
PMA disse:  Ops! O nome do medicamento não é válido.
PMA disse:  Não quererá dizer enalapril?
User diz (Sim ou não): sim
~~ Medicamento seleccionado: ENALAPRIL__3.4.2.1. ~~
PMA disse:  a dosagem recomendada de enalapril é 10,0 miligramas a 40,0 miligramas.
User diz:  obrigada pela ajuda
PMA disse:  Estou ao seu dispor. Em que mais posso ajudá-lo?
User diz:  qual é a frequência de enalapril
~~ Medicamento seleccionado: ENALAPRIL__3.4.2.1. ~~
PMA disse:  a frequência prescrita de enalapril é 1 por 1 dia.
User diz:  qual é a via de enalapril
~~ Medicamento seleccionado: ENALAPRIL__3.4.2.1. ~~
PMA disse:  a via recomendada de enalapril é oral.
User diz:  qual é a administração de enalapril
~~ Medicamento seleccionado: ENALAPRIL__3.4.2.1. ~~
PMA disse:  a administração recomendada de enalapril é oral.
User diz:  existem restrições para enalapril
~~ Medicamento seleccionado: ENALAPRIL__3.4.2.1. ~~
PMA disse:  o medicamento prescrito enalapril não apresenta quaisquer restrições!
User diz:  existem restrições para bisoprolol
PMA disse:  O medicamento bisoprolol não faz parte da sua prescrição médica.
PMA disse:  Não quererá dizer enalapril?
User diz (Sim ou não): não
ePMA disse:  O que deseja perguntar então?
User diz:  existem restrições para paracetamol
PMA disse:  Não percebi o que disse. Pode repetir?
User diz:  existem restrições para paracetamol
~~ Medicamento seleccionado: PARACETAMOL__2.10. ~~
PMA disse:  o medicamento prescrito paracetamol apresenta restrições entre 500,0 miligramas a 1000,0 miligramas.
User diz:  quais são as posologias existentes de enalapril
~~ Medicamento seleccionado: ENALAPRIL__3.4.2.1. ~~
PMA disse:  as posologias registadas de enalapril são:
PMA disse:  posologia 2
PMA disse:  posologia 1
PMA disse:  posologia 0
User diz:  quais são as reacções adversas de enalapril
~~ Medicamento seleccionado: ENALAPRIL__3.4.2.1. ~~
PMA disse:  Ops! A palavra reacções não é válida. Por favor repita.
User diz:  quais são as reações adversas de enalapril
~~ Medicamento seleccionado: ENALAPRIL__3.4.2.1. ~~
PMA disse:  o medicamento prescrito enalapril não tem quaisquer reações!
User diz:  quais são as reações adversas de paracetamol
~~ Medicamento seleccionado: PARACETAMOL__2.10. ~~
PMA disse:  as reações adversas de paracetamol são:
PMA disse:  descritas doenças hematológicas doenças
PMA disse:  Erupções cutâneas
PMA disse:  pancreática
PMA disse:  insuficiencia renal
PMA disse:  intoxicação o risco de insuficiencia hepática
User diz:  porque estou a tomar enalapril
PMA disse:  Podia fazer o favor de repetir?
User diz:  Porque razão estou a tomar enalapril
~~ Medicamento seleccionado: ENALAPRIL__3.4.2.1. ~~
PMA disse:  o medicamento prescrito enalapril não tem informação registada suficiente para responder a esta pergunta!
User diz:  Porque razão tenho de tomar paracetamol
~~ Medicamento seleccionado: PARACETAMOL__2.10. ~~
PMA disse:  as razões que justificam a toma de paracetamol são:
PMA disse:  dor ligeira moderada
PMA disse:  febre
User diz:  quais os medicamentos que o médico receitou?
PMA disse:  O seu médico receitou:
PMA disse:  enalapril
PMA disse:  eplerenona
PMA disse:  paracetamol
```

Figure A.1: Screenshot with an example of PMA interaction during the second stage of development. (User utterances in green)

```
PMA_pt [Java Application] /usr/lib/jvm/java-6-openjdk-i386/bin/java (May 20, 2013 3:15:57 PM)
User diz:  posso tomar bisoprolol
PMA disse:  Não pode tomar bisoprolol porque não faz parte da sua prescrição médica.
PMA disse:  Tem-se sentido bem?
User diz (Sim ou não): não
PMA disse:  Tem tido falta de ar?
User diz (Sim ou não): não
PMA disse:  Tem tido dores no peito?
User diz (Sim ou não): não
PMA disse:  Tem tido cansaço intenso?
User diz (Sim ou não): não
PMA disse:  Tem tido aumento de peso superior a 2 kg por semana?
User diz (Sim ou não): não
PMA disse:  Tem tido tosse constante?
User diz (Sim ou não): não
PMA disse:  Tem tido pés e pernas inchados?
User diz (Sim ou não): não
PMA disse:  Porque não se sente bem então?
User diz: Dói-me muito a cabeça.
PMA disse:  Os seus problemas vão ser reportados por mensagem ao médico.
PMA disse:  Atenção! Deve ter cuidado com a alimentação manter um estilo de vida saudável e controlar o seu peso todas as semanas.
```

Figure A.2: Screenshot with a use case of PMA medical inquiry during the second stage of development. (User utterances in green)

Table A.II: User information and PMA performance data for the two tasks executed during the usability test.

	User	A	B	C	D	E	F	G	H	I	J	K	*Averages*
	Gender	F	F	M	M	M	M	F	M	M	F	M	
	Age	22	22	30	24	24	22	26	30	26	24	24	24.9
Task I	**Goal**	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	
Task I	**Time elapsed (min)**	5	3	5	4	5	3	1	7	4	2	2	3.7
Task I	**Nº of utterances**	5	6	5	4	5	4	3	4	3	6	4	4.5
Task II	**Goal**	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	
Task II	**Time elapse (min)**	3	4	1	2	1	4	1	2	3	3	4	2.5
Task II	**Nº of utterances**	5	6	3	2	1	3	1	2	5	5	6	3.5
	Daily medicines ? (*number*)	No	Yes (1)	No	No	Yes (1)	Yes (2)	No	Yes (5)	No	No	No	
	Work with seniors? (*years*)	No	No	Yes (½)	No	No	Yes (½)	Yes (2)	Yes (3)	No	No	No	

Table A.III: Average scores and standard deviations for each of the 16 statements of the usability questionnaire with the system property they evaluate.

	Question no.	*Average score (0-4)*	*Dev.*
Usability	1	2	0.93
	2	3	0.81
	3	2	0.79
	7	2	0.66
	12	3	0.91
Coherence	4	2	0.60
	5	3	0.69
Naturalness	6	3	0.68
Information quality	8	4	0.40
	9	3	0.83
	10	3	0.60
	11	3	0.91
Usability for seniors	13	3	0.53
	14	3	1.01
	15	3	0.79
	16	3	0.99

Appendix B: Usability questionnaire

Questionnaire - *based on the System Usability Scale questionnaire (SUS)*

User: **Age:** **M** ☐ **F** ☐

Do you take medication regularly?

If YES, how many pills per day?

In the rating sheet below, please circle the number that most closely matches your feelings about the Personal Medication Advisor (PMA) as a user.

1. In general, I found the system unnecessarily complex.

Strongly disagree	1	2	3	4	Strongly agree

2. In general, I thought the system was easy to use.

Strongly disagree	1	2	3	4	Strongly agree

3. I think that I would need the support of a technical person to be able to use this system.

Strongly disagree	1	2	3	4	Strongly agree

4. I thought there was too much inconsistency in the dialogue.

Strongly disagree	1	2	3	4	Strongly agree

5. The dialog I had with the system was coherent.

Strongly disagree	1	2	3	4	Strongly agree

6. I felt the dialogue with the system was natural.

Strongly disagree	1	2	3	4	Strongly agree

7. I found the system very awkward to use.

Strongly disagree	1	2	3	4	Strongly agree

8. The information provided by the system is easy to understand.

Strongly disagree	1	2	3	4	Strongly agree

9. The system gives error messages that clearly tell me how to fix problems.

Strongly disagree	1	2	3	4	Strongly agree

10. I was able to get useful information using this system.

Strongly disagree	1	2	3	4	Strongly agree

11. I quickly could get the information I needed.

Strongly disagree	1	2	3	4	Strongly agree

12. I felt very confident using the system.

Strongly disagree	1	2	3	4	Strongly agree

Please answer the next questions having a **senior user** in mind:

13. The interface of this system is pleasant.

Strongly disagree	1	2	3	4	Strongly agree

14. Older people would learn to use this system very quickly.

Strongly disagree	1	2	3	4	Strongly agree

15. This system would help senior patients to understand and manage their medication regimen.

Strongly disagree	1	2	3	4	Strongly agree

16. For those who follow strict medications regimens, I think they would like to use this system frequently.

Strongly disagree	1	2	3	4	Strongly agree

Do you usually work for or with senior people?

If YES, how long have you been working for or with them?