

Linguagens Visuais para Aplicações Médicas

Nuno Carvalho, Rui Couto, J. Artur Vale Serrano

Resumo- Este artigo descreve uma interface gráfica para uma aplicação médica.

Esta interface utiliza linguagens visuais e tem como função permitir uma melhor gestão das consultas externas de uma instituição hospitalar através de um mecanismo de agendamento.

Abstract – This article describes a graphical interface for a medical application.

The interface uses visual languages and its main function is the management of an healthcare institution. For this we use a scheduling system.

I. INTRODUÇÃO

As linguagens visuais estão a ser cada vez mais usadas pela indústria de *software*, devido à sua forte componente gráfica.

A utilização de cor e de grafismos apelativos, aliados à funcionalidade acrescida deste tipo de interface de utilizador, torna-o preferencial quando comparado com interfaces baseadas puramente em texto.

O trabalho aqui descrito enquadrou-se nestes princípios: construir uma interface para uma aplicação médica recorrendo a linguagens visuais, que fosse ao mesmo tempo atraente, inovadora e sobretudo funcional.

Sabendo que as instituições de saúde em Portugal ainda não aderiram completamente às novas tecnologias de informação, achamos fundamental que a interface a desenvolver tivesse as características atrás citadas bem presentes, de modo a que a mudança para os novos sistemas de informação seja feita de forma a minimizar o impacto nos utilizadores.

Este trabalho foi baseado em estudos realizados no âmbito de projectos a nível europeu. Nomeadamente, todo este sistema está assente numa base de dados construída segundo a norma HISA¹ (que propõe um modelo genérico para a construção de sistemas de informação hospitalares), da qual extraímos apenas a informação necessária para executarmos o nosso trabalho. Esta base de dados será articulada com um sistema de regras, que verifica a validade das operações a efectuar.

A parte que fizemos diz respeito à agenda de consultas, estando também englobada a parte referente aos recursos humanos, neste caso apenas os médicos.

II. OBJECTIVOS

O principal objectivo do nosso trabalho foi criar uma interface gráfica para uma aplicação médica, que tivesse como função permitir uma melhor gestão das consultas externas de uma instituição hospitalar.

Este trabalho destina-se a ser integrado num projecto mais alargado, pelo que devemos sempre tomar decisões que tornem fácil a sua posterior adaptação para outros sistemas.

Os requisitos para esta aplicação foram:

- a marcação e eliminação de consultas externas, tendo em conta os recursos hospitalares a estas associados;
- a gestão do horário de disponibilidades de cada médico da instituição hospitalar;
- a visualização (em vários tipos de agenda) do mapa de marcações efectuadas;
- a visualização dos dados dos pacientes e médicos quando solicitados;
- a inserção de novos pacientes e médicos.

O armazenamento dos dados foi feito através de uma base de dados contendo toda a informação relevante para o funcionamento do sistema. Esta base de dados será descrita mais à frente.

III. DESENVOLVIMENTO DO TRABALHO

Tendo por base a norma HISA, desenhámos a base de dados que serve de suporte ao sistema desenvolvido.

Para o desenho da base de dados, utilizámos a ferramenta CASE *Paradigm Plus*, e para a sua implementação utilizámos o *Microsoft Access 97*.

O *Paradigm Plus*, também foi utilizado para o desenho de classes e para a especificação de diálogos da interface de utilizador.

Na implementação da interface, foi usada a linguagem Java, suportada pela ferramenta Borland JBuilder 2.

As componentes base de dados e interface de utilizador são descritas nas secções abaixo.

¹ "Healthcare Information System Architecture Part 1 (HISA) HealthCare Middleware Layer", Final Draft 2, CEN/TC 251, <http://www.lhto.be/ikb/standards/centc251/hisa/index.html> , CEN – Comité Européen de Normalisation, March, 1997.

A. Base de dados

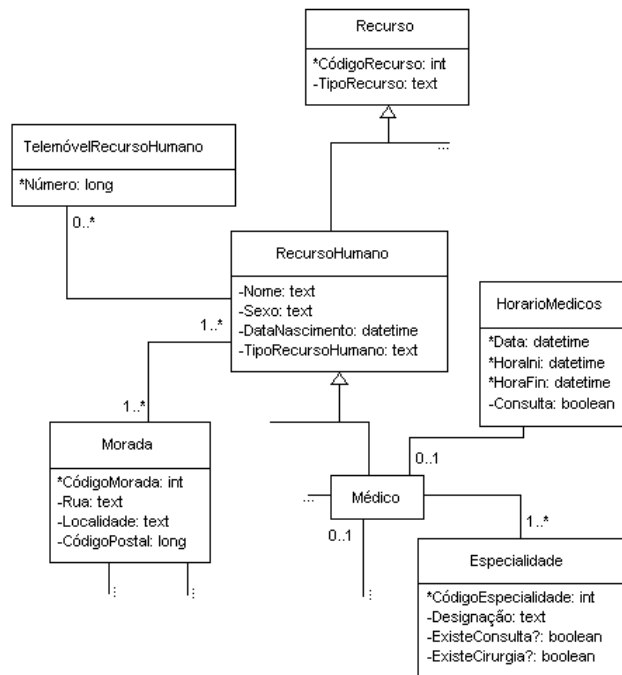


Fig. 1-Extracto do diagrama de classes da base de dados.

A norma HISA propõe um modelo genérico para a construção de Sistemas de Informação Hospitalares (SIH), do qual extraímos apenas a informação necessária para a nossa base de dados.

Na figura 1, é mostrado um extracto do diagrama de classes desta base de dados, onde utilizámos como linguagem de modelação de objectos os diagramas de classes da UML. Neste diagrama podemos ver que a tabela RecursoHumano possui como campos o nome, sexo, data de nascimento e o tipo de recurso humano de um determinado indivíduo que pertença à instituição hospitalar. Cada um destes indivíduos, pode ter zero ou mais telemóveis, mas tem que ter sempre referência para pelo menos uma morada. Os médicos são um tipo de recurso humano, e têm que estar obrigatoriamente referenciados na tabela Especialidade. Esta, no entanto pode conter várias referências a entradas da tabela Médico.

Os dados armazenados por esta base de dados, podem ser divididos em dois grupos:

- dados pré-definidos, que compreendem:
 - recursos humanos e materiais;
 - especialidades;
 - actividade e tipo de actividade.
- dados a definir no local, que compreendem:
 - informação referente aos pacientes.

Especificação de requisitos:

Pretende-se com esta base de dados manter a informação actualizada sobre recursos humanos e materiais, bem como as actividades exercidas nas diversas áreas da

instituição, as várias especialidades existentes, e também informação relativa aos pacientes.

Relativamente à informação sobre recursos humanos e materiais, a instituição está interessada em saber os tipos de recursos de que dispõe, tais como médicos, enfermeiros, auxiliares, técnicos, salas e equipamento diverso. Para os recursos humanos, interessa saber o nome, código interno, e a(s) respectiva(s) morada(s) caracterizadas pela rua, localidade e código postal. Para poder contactar estes recursos rapidamente, pretende a instituição conhecer também o(s) telefone(s) e fax(es) da(s) morada(s) onde se encontra(m) instalado(s), assim como o(s) telemóvel(eis) e e-mail(s) de cada recurso.

No caso do recurso médico, a instituição deseja também saber o seu estatuto.

No que respeita aos recursos auxiliar e técnico, pretende-se saber ainda a secção a que dizem respeito.

Em relação ao recurso equipamento pretende-se saber apenas o nome e o seu código interno.

Relativamente ao recurso sala pretende-se saber o nome, o seu código interno, a sua localização, o seu estatuto e se está ou não ocupada.

Em relação às especialidades existentes naquela instituição de saúde pretende-se ter informação sobre a existência ou não de cirurgias ou consultas para cada especialidade.

No que diz respeito às actividades, pretende-se saber que tipos de actividades existem naquela instituição. O tipo de actividade é composto por um número de quatro códigos, que caracterizam essa actividade de acordo com uma tabela predefinida pela Ordem dos Médicos.

Há ainda dois parâmetros, definidos pela Ordem dos Médicos, para o tipo de actividade, que são o C (equipamento necessário) e o K (componente mão-de-obra).

Relativamente à informação sobre os pacientes, a instituição está interessada em saber o seu nome, o sexo, data de nascimento, data de falecimento (se for o caso) e a(s) respectiva(s) morada(s) caracterizadas pela rua, localidade e código postal. Para poder contactar os pacientes rapidamente, pretende a instituição conhecer também o(s) telefone(s) e fax(es) da(s) morada(s) onde se encontra(m) instalado(s), assim como o(s) telemóvel(eis)/e-mail(s) de cada paciente.

Pretende-se com esta base de dados, implementar um sistema de marcação de consultas que permita a escolha do(s) médico(s), da data de início e fim, da sala e a actividade.

Deverá ser possível também, alterar a agenda, apagando ou acrescentando marcações.

Na altura da marcação de uma consulta, o sistema deverá rejeitar todas as marcações que se tentem fazer nos dias em que o médico não esteja seleccionado para as consultas externas.

B. Interface de Utilizador

A definição da interface de utilizador, foi dividida em várias fases.

Análise:

Começou-se por analisar os dados recolhidos numa unidade de saúde, para se ter conhecimento real do seu funcionamento.

De seguida, e com base nesta análise elaborámos um documento hipermédia baseado em HTML, que descreve os traços gerais da interface a criar.

Desenho do diagrama de classes da interface:

Nesta fase do trabalho, começámos por estruturar a interface em termos de classes.

Para cada classe foi definido um conjunto de atributos e métodos necessários à realização de uma ou mais funções.

O diagrama de classes foi desenhado agrupando as classes de acordo com atributos e métodos comuns, e as relações de dependência entre elas.

Como linguagem de modelação de objectos usámos os diagramas de classes da *UML*.

Como exemplo temos:

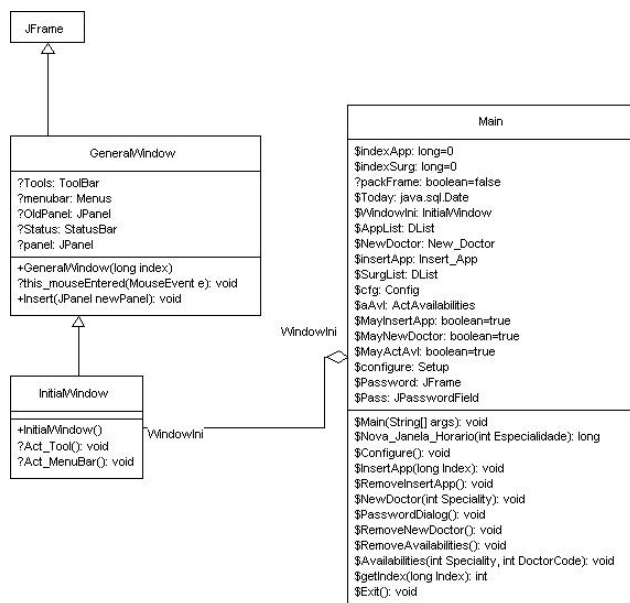


Fig. 2 – Extracto do diagrama de classes da interface.

A figura acima representa uma pequena parte do diagrama de classes da interface. A classe GeneralWindow deriva da classe JFrame, que é uma classe pré-definida e representa uma janela. A classe derivada (GeneralWindow), herda os atributos e métodos da classe mãe (JFrame), ficando assim mais completa. A classe InitialWindow, por sua vez, deriva da classe GeneralWindow, e é a classe responsável pela janela

inicial do programa. A classe Main possui uma instância da classe InitialWindow (WindowIni), daí a sua associação com esta e é a partir da classe Main que se inicia o programa.

Especificação de diálogos:

O desenho da interface foi guiado por um conjunto de diagramas elaborados previamente. Estes diagramas ilustram as várias etapas a ser executadas pelo utilizador sobre a interface (diálogos).

Mais uma vez, como linguagem de modelação de usámos a *UML*, desta feita os diagramas de interacção.

Exemplo:

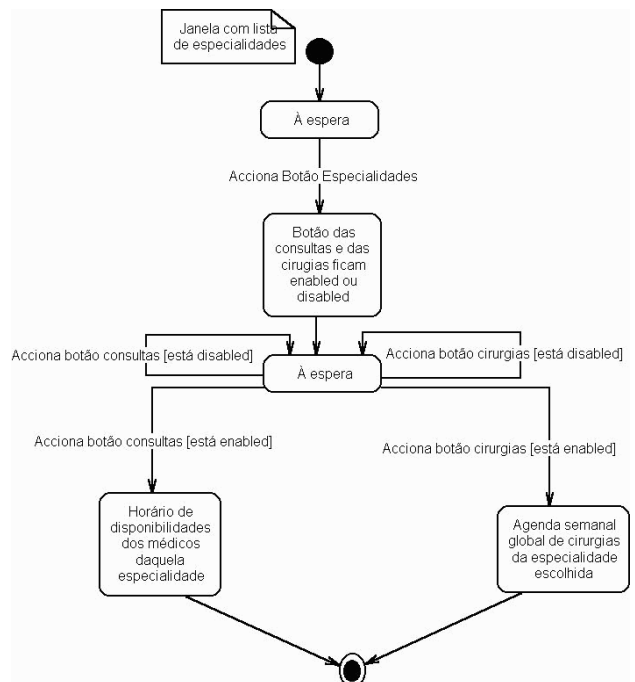


Fig. 3 – Diagrama de especificação de diálogo da escolha de especialidades.

Este diagrama diz respeito à escolha de especialidades.

Quando se está a visualizar a lista de especialidades, a aplicação espera que o utilizador accione um botão referente a uma especialidade. De acordo com a especialidade escolhida, os botões referentes às consultas e às cirurgias ficarão no estado “disponível” ou “indisponível”.

Quando o utilizador accionar um destes botões será executada uma dada acção, desde que esse botão esteja no estado disponível.

No caso do botão das consultas, será mostrado o horário de disponibilidades dos médicos da especialidade escolhida. No caso de o utilizador ter accionado o botão das cirurgias, será mostrada a agenda global semanal, com a indicação das cirurgias marcadas.

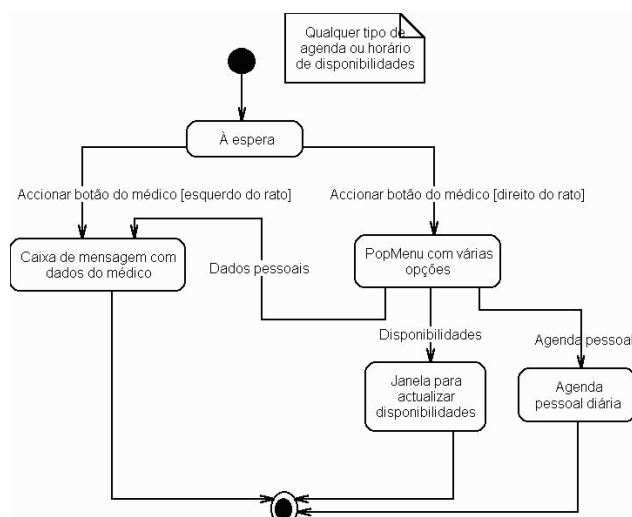


Fig. 4 – Diagrama de especificação de diálogo dos botões referentes aos médicos.

IV. DESCRIÇÃO DO SISTEMA

Funcionalidade da interface

A funcionalidade oferecida pela aplicação é definida pelos diagramas de diálogo, descritos anteriormente. A informação a visualizar durante a interacção com o sistema é apresentada de seguida.

Informação a visualizar

A informação a visualizar é transmitida pelos diversos tipos de agenda, cada tipo revela-nos um ou vários níveis de informação. Alguns dos tipos de agenda, assim como os níveis de informação por eles transmitida são:

“O horário de disponibilidades” - É um horário semanal (fig. 5) referente a uma especialidade, onde são mostradas as horas em que os médicos dessa especialidade estão disponíveis para dar consultas externas.



Fig. 4 – Horário de disponibilidades dos medicos.

“A agenda diária de um médico” - Deste tipo de agenda (fig. 6) pode-se retirar a informação relativa à disponibilidade do médico a que pertence a agenda em termos de horas. A disponibilidade é representada pelas células a uma cor distinta, ou seja só se podem marcar consultas nos intervalos dessa cor*.

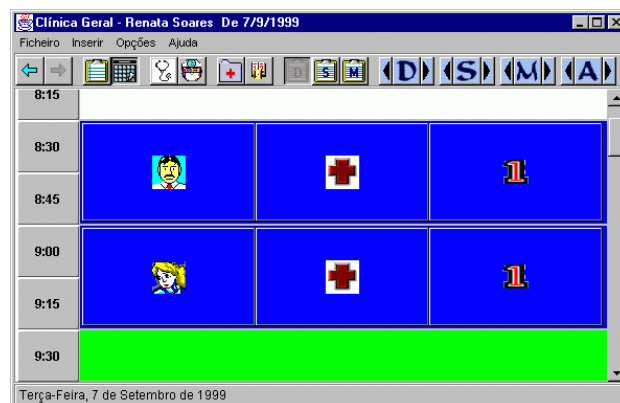


Fig. 6 – Agenda diária de um médico.

As consultas são representadas por um painel de cor mais escura, que contém três botões. Um botão que indica o sexo (imagem do botão) e o nome do paciente, outro botão que indica o número e o tipo de sala da consulta e um último botão que indica o número e tipo de consulta. Qualquer um destes dados se sabe colocando o ponteiro do rato sobre um dos botões da consulta.

“A agenda mensal de um médico” - Deste tipo de agenda (fig. 7) pode-se retirar a informação relativa à disponibilidade do médico a que pertence a agenda em termos de dias. Mais uma vez a disponibilidade é assinalada por uma coloração distinta das células respectivas, ou seja os dias que estiverem desta cor são os dias que o médico está disponível para dar consultas.

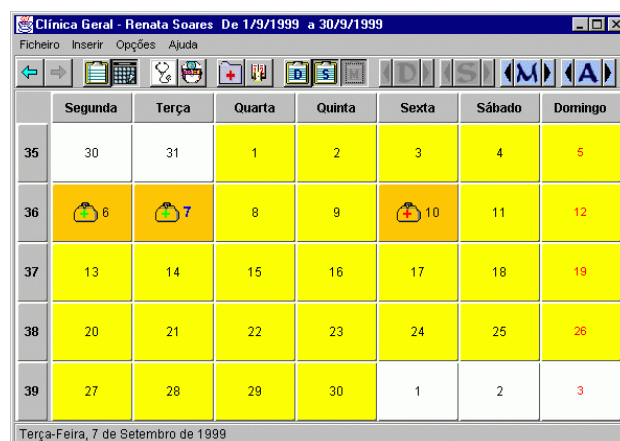


Fig. 7 – Agenda mensal de um médico.

* Devido ao facto de esta revista ser a preto e branco, não é possível mostrar as cores usadas pela interface construída, que lhe conferem maior legibilidade.

Os dias em que o médico já tem consultas marcadas aparecem com um ícone, que é uma mala. A mala com uma cruz clara indica que ainda se podem marcar mais consultas para o médico a que pertence a agenda. A mala com uma cruz escura indica que já não se podem marcar mais consultas para o médico a que pertence a agenda.

Marcação de consultas

A marcação de consultas só pode ser efectuada na agenda diária do médico de uma especialidade. Para iniciar o processo de marcação, deve-se accionar com o botão esquerdo do rato no intervalo de tempo em que quer que comece a consulta. Só se pode escolher os intervalos de tempo em que o médico está disponível. Depois aparecerá a janela inicial de marcação de consultas (fig. 8), em que será especificada a data e hora dos mesmos.

Fig. 8 – Janela inicial de marcação de consultas.

Ao accionar-se o botão **Seguinte** aparecerá outra janela para selecção do paciente (fig. 9).

Fig. 9 – Janela para selecção do paciente.

Pode fazer-se uma pesquisa para ver se o paciente já está registado na unidade de saúde, através do número, último nome, primeiro nome ou ambos.

Se o paciente não for encontrado, surgirá uma mensagem de aviso para o efeito, neste caso, pode optar por inserir um novo paciente, ou fazer nova pesquisa. Se existir mais do que um paciente com o primeiro ou último nome

pesquisado, então pode-se seleccionar qual o paciente pretendido.

Finalmente, accionando o botão **Seguinte** aparecerá uma última janela (fig. 10) para inserir o número da sala e seleccionar o tipo de consulta (1ª consulta se for a primeira consulta do paciente com o médico, especialista se se tratar de uma consulta para um especialista).

Fig. 10 – Janela final de marcação de consultas.

Outras funcionalidades

Além das funcionalidades já descritas, a interface possui outras que também são relevantes, das quais passamos a apresentar.

- Visualização dos dados pessoais de um paciente

Para ver os dados pessoais de um paciente, basta accionar o botão direito do rato num dos botões referentes a um paciente numa das consultas. Será então mostrada uma janela (fig. 11) com os dados pessoais do paciente.

Fig. 11 – Dados relativos a um paciente.

- Ficha médica de um paciente

Pode-se editar ou visualizar a ficha médica de um paciente. Será aberto um documento com a extensão *txt* numa janela do *notepad*. Esse documento apresentará toda a informação que os médicos tenham escrito sobre o paciente em causa. Se ainda não existir a ficha médica do

paciente, então será criado e aberto um novo documento com o cabeçalho com a identificação do paciente.

- Inserção de um novo paciente

A inserção de um novo paciente, apenas é possível aquando da marcação de consultas. Para esta operação será mostrada uma janela (fig. 12), onde se deve introduzir os dados do paciente.

Fig. 12 – Janela de inserção de um novo paciente.

- Visualização dos dados pessoais de um médico

Pode-se também ver os dados pessoais de um médico, através dos botões referentes aos médicos (tal como exemplificado no diagrama de diálogos da figura 4) que aparecem nas várias agendas. Será então mostrada uma janela (fig. 13) com os dados pessoais do médico.

Fig. 13 – Dados relativos a um médico.

- Alteração do horário de disponibilidades dos médicos

É possível alterar o horário de disponibilidades dos médicos através duma janela específica (fig.14) actuando sobre os botões dos médicos que aparecem nas agendas, ou através da barra de menus. Se estiver numa agenda pessoal dum médico, então apenas é possível alterar o horário de disponibilidades desse médico. Se estiver num

horário de disponibilidades ou numa agenda global, então poderá alterar o horário de disponibilidades de qualquer um dos médicos da especialidade seleccionada.

Fig. 14 – Janela de actualização do horário de disponibilidades dos médicos.

- Inserção de um novo médico

Para se inserir um novo médico, deve-se seleccionar a opção **Inserir | Médico** na barra de menus. Será pedida a palavra-chave, que só o administrador do sistema possui. Depois de introduzida correctamente, aparecerá uma janela (fig. 15) na qual se podem inserir os dados relativos ao médico que se pretende inserir.

Fig. 15 – Janela de inserção de um novo médico.

Facilidade de configuração

De modo a permitir uma maior liberdade na configuração da aplicação, decidimos incluir um ficheiro que contém os principais dados configuráveis (língua, ícones, duração mínima de consultas, mínimo intervalo de tempo em que o médico está disponível, hora inicial e final do serviço). Para isso basta editar directamente o ficheiro Livam.ini que está armazenado na directoria Windows. Este ficheiro contém uma listagem das características do programa que podem ser configuradas.

Justificação de algumas das opções tomadas

Em relação à janela com que se inicia o programa, optámos por escolher uma janela (fig. 16) contendo a lista de especialidades existentes na unidade de saúde.

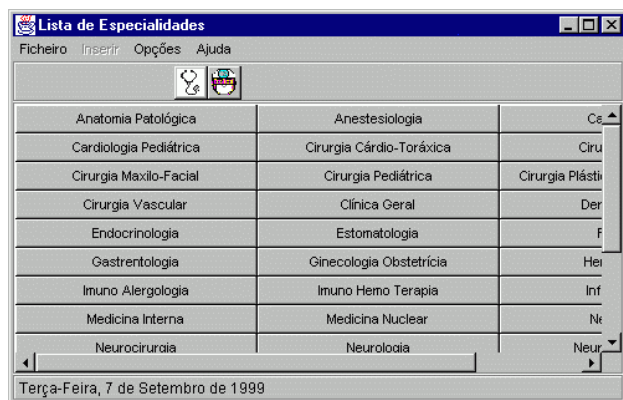


Fig. 16 – Janela inicial da interface.

Deste modo, o utilizador pode facilmente escolher com qual das especialidades pretende trabalhar.

Como nem todas as especialidades possuem consultas ou cirurgias, esta janela apresenta dois botões que indicam se a especialidade seleccionada possui ou não consultas ou cirurgias.

Se o utilizador se decidir pelas consultas, aparecerá uma janela com o horário de disponibilidades dos médicos da especialidade seleccionada, para a semana em que está inserido o dia actual (tal como se pode ver no diagrama de diálogo da figura 3). Deste modo o utilizador tem uma visão geral dos médicos que estão disponíveis para consultas nessa semana.

Para facilitar a visualização de consultas, decidimos implementar três formatos de agenda (diária, semanal, mensal) permitindo ao utilizador ter três níveis de informação.

Tendo em conta que aplicação pode crescer para uma utilização em rede, decidimos implementar dois tipos de agenda:

- Agenda global, que permite a visualização da informação referente a todos os médicos de uma dada especialidade;
- Agenda pessoal, que apenas contém a informação referente a um médico de uma dada especialidade. Deste modo está garantida alguma privacidade, pois cada médico apenas poderá aceder à sua agenda pessoal.

Nos formatos semanal e diário da agenda global, optámos por reservar uma linha ou coluna para cada médico que tenha consultas num determinado dia, por forma a termos a informação mais bem organizada, permitindo ao utilizador uma melhor percepção da mesma.

Inicialmente tínhamos pensado usar um mecanismo de *drag-and-drop* para fazer a marcação de consultas. No entanto este mecanismo era pouco funcional, pelo que optámos por fazer aparecer uma sequência de janelas que

permitia ao utilizador inserir os diversos dados por etapas. Deste modo o processo de marcação de uma consulta é funcional e não é maçador.

A pesquisa do paciente na base de dados, para marcar uma consulta pode ser feita por:

- número;
- primeiro e/ou último nome.

Deste modo o utilizador tem vários modos de pesquisa ao seu dispor para prevenir eventuais faltas de informação.

V. CONCLUSÕES E TRABALHO FUTURO

Depois de termos elaborado este trabalho, podemos afirmar que ele se revelou importante por várias razões, das quais destacamos o facto de termos iniciado e concluído um projecto, participando em todas as fases da sua concepção.

Durante as várias fases do trabalho, aprendemos novos conceitos e trabalhamos com novas ferramentas de desenvolvimento de software (*JBuilder2*, *Paradigm Plus*), que até então nos eram desconhecidas.

A fase de desenho revelou-se importante para percebermos melhor os conceitos associados à modelação de objectos, e porque a estrutura da interface, em termos de classes, fica definida simplificando o trabalho seguinte. Algumas das opções tomadas nesta fase, foram alteradas, porque verificámos que não eram as melhores para implementar o que pretendíamos.

A especificação de diálogos entre o utilizador e a interface, é uma fase que deve ser tida em conta, porque permite definir e esclarecer os passos de uma dada acção.

Em relação à fase de desenvolvimento de software, aprendemos a programar em Java. Esta linguagem mostrou-se, de fácil utilização e eficiente, apesar de a versão que usámos (JDK 1.1.6) ainda conter alguns "bugs".

A ferramenta que utilizámos para desenvolvimento em Java, o *Borland JBuilder2*, apesar de possuir uma interface bastante amigável, denota ainda alguns problemas em vários aspectos. De referir que os meios de que dispunhamos (dois PC's equipados com processadores Pentium a 133MHz e 64MB de memória RAM) se revelaram largamente insuficientes para desenvolver aplicações deste tipo.

No que diz respeito à base de dados pudemos perceber melhor os conceitos associados ao desenho e implementação de uma base de dados relacional, utilizando o *Microsoft Access 97*.

A aplicação desenvolvida destina-se apenas à gestão de consultas externas. No entanto, foi desenvolvida a pensar em futuras versões bem mais completas.

As próximas versões, podem facilmente:

- englobar a gestão de cirurgias, visto que já existem algumas janelas preparadas para esse efeito, e já existem ícones destinados às cirurgias;
- permitir configurar o ambiente de trabalho, alterando todos os ícones, usando um mecanismo de selecção

baseado em janelas, ou através de edição directa num ficheiro;

- usar outras línguas, para além do inglês e francês, ou alterar completamente os textos (de forma semi-automática), visto que estas frases e palavras são adquiridas em “*run-time*”, e estão guardados num ficheiro por cada língua.

VI. REFERÊNCIAS

- [1] Shneiderman, Ben, Designing the User Interface: Strategies for Effective Human-Computer Interaction, Third Edition, Addison-Wesley, 1998.
- [2] Fowler, Martin, Scott, Kendall, UML distilled: applying the standard object modeling language, Addison-Wesley, 1997.
- [3] Horton's, Ivor, Beginning Java, First Edition, Wrox Press Ltd. 1997.
- [4] Weiner, Scott R., Asbury, Stephen, Programming with JFC, John Wiley & Sons, Inc., 1998.
- [5] Reese, George, Database Programming with JDBC and JAVA, First Edition, O'Reilly & Associates, Inc., 1997.
- [6] Eckel, Bruce, Thinking in Java, Prentice Hall, 1998.
- [7] Serrano, J. Artur, "The Use of Semantic Constraints on Diagram Editors", presented at 11th International IEEE Symposium on Visual Languages, Darmstadt, Germany, 5-6 September 1995.
- [8] Serrano, J. Artur and Welland, Ray, "VC ' – A Formal Language for the Specification of Diagrammatic Modelling Techniques", to be published in journal Information and Software Technology, Elsevier, 1998.
- [9] Serrano, J. Artur and Jacinto, Marta, "A Semantic-Constraints Based Scheduler for Healthcare Information Systems", presented at EIS'99 - International Conference on Enterprise Information Systems, Setúbal, Portugal, 1999.
- [10] JavaWorld, 1999, : <http://www.javaworld.com/>
- [11] CodeGuru, 1999, "Java", <http://codeguru.developer.com/java>
- [12] Java, 1999, (online): <http://java.sun.com/>
- [13] Prague, Cary N., Irwin, Michael R., "Access 97 Bible", IDG Books WorldWide, Inc., 1997.
- [14] Harner, Yuri, Loftin, Michele, Salemie, Emely, "Paradigm Plus, User's Guide", Platinum Technology, 1997.