

Mestrado em Engenharia Informática

Estágio

Relatório Final

Visão e Análise Temporal do Processo Clínico

Celso Rafael Clara Mendes

celsom@student.dei.uc.pt

Orientador do DEI-FCTUC:

Professor Doutor César Teixeira

Orientador da *MedicineOne*:

Engenheiro Nuno Teixeira

Coimbra, 1 de Julho de 2016



FCTUC DEPARTAMENTO
DE ENGENHARIA INFORMÁTICA
FACULDADE DE CIÊNCIAS E TECNOLOGIA
UNIVERSIDADE DE COIMBRA

Mestrado em Engenharia Informática

Estágio

Relatório Final

Visão e Análise Temporal do Processo Clínico

Celso Rafael Clara Mendes

celsom@student.dei.uc.pt

Júris:

Professor Doutor Paulo Carvalho

Professor Doutor Marco Vieira

Coimbra, 1 de Julho de 2016



FCTUC DEPARTAMENTO
DE ENGENHARIA INFORMÁTICA
FACULDADE DE CIÊNCIAS E TECNOLOGIA
UNIVERSIDADE DE COIMBRA

Resumo

Atualmente, processos clínicos de utentes podem ser consideravelmente extensos, contendo desde informação relativa a exames realizados até a problemas de saúde que possam existir. Atendendo à extensão dos dados que podem existir, a sua análise e visualização torna-se difícil. Adicionalmente, para além de a leitura e interpretação se tratar de um processo moroso, a informação pode ainda estar em diferentes documentos/localizações, por vezes em relatórios e sem se encontrar totalmente formatada. Uma possível solução para facilitar, e em geral melhorar, o processo é a utilização de uma visualização que reduza o esforço cognitivo e permita mais rapidamente identificar e extrair a informação relevante e tirar conclusões.

Neste projeto tenho como objetivo estudar e implementar a solução identificada, para que um profissional de saúde possa rapidamente e com qualidade visualizar a informação clínica e facilmente entender as necessidades do utente.

Na linha do tempo o profissional clínico terá ao seu dispor diferentes formas de visualizar a informação, desde a visualização da situação clínica atual, análise do histórico clínico e até mesmo entender as necessidades clínicas que o utente possa vir a ter no futuro, tais como, vacinas, exames periódicos, etc..

De forma a enriquecer os dados disponíveis também se pretende fornecer ao profissional clínico informação sobre a possibilidade de um paciente poder vir a contrair uma determinada patologia.

Neste documento é possível observar o trabalho desenvolvido neste âmbito e as vantagens que este projeto trará para a área em questão.

Palavras-chave: *Timeline, Patient Timeline, Medical Timeline, Clinical Decision Support System, Lifelines, Data Visualization, Information Visualization, pattern recognition in healthcare*

Abstract

Patients' clinical processes can be considerably extensive, containing information from clinical test results to diagnosed health issues. Taking in account the volume of data that might accumulate, visualising and analysing it becomes a complex task. On top of how slow the process of reading and interpreting the data is, the relevant information might be spread across distinct documents/locations and in unstructured formats. A possible solution to ease, and generally improve, this process is using a better visualization aimed at reducing the cognitive effort and speeding the identification and extraction of the relevant information as well as the conclusions that might follow.

The objective of the project is the research and implementation of the proposed solution, empowering medical personnel with high quality and fast means of clinical information acquisition and analysis. A direct benefit is giving the medical professional a clear view of the patient's state and help him better understand their needs.

On the timeline, the clinical professional will have at their disposal a myriad of ways to visualize not only the current clinical data but the past history as well. Additional, possible future needs, such as vaccinations, periodic clinical tests, etc.

With the objective of enriching the data available to the healthcare professionals, the visualization, will also contain indication of pathologies a patient might contract as a result of their medical history and lifestyle.

This document details the efforts undertaken in this scope as well as exposing the resulting advantages to the healthcare field.

Keywords: *Timeline, Patient Timeline, Medical Timeline, Clinical Decision Support System, Lifelines, Data Visualization, Information Visualization, pattern recognition in healthcare*

Agradecimentos

Em primeiro lugar queria agradecer à *MedicineOne* e ao João Miguel por me terem disponibilizado este desafio e pelo seu profissionalismo e disponibilidade. Tenho a deixar um especial agradecimento ao meu orientador, Nuno Teixeira, pela sua amizade, disponibilidade, confiança, paciência e orientação ao longo deste projeto. Não posso também deixar de agradecer ao Nuno Pereira pelo tempo que disponibilizou para me ajudar e esclarecer as dúvidas que foram surgindo. Agradeço, também, ao Luís Pinto, à Renata Rojão, ao João Araújo, ao Bruno Almeida, ao Marco Tinoco, ao Bruno Costa e aos demais elementos da *MedicineOne* pelo o apoio prestado ao longo deste período que estive inserido nesta equipa.

Tenho também a agradecer ao Professor Doutor César Teixeira pela excelente orientação que me foi transmitindo ao longo deste estágio, pela disponibilidade, pela sua atenção e pelo seu apoio ao longo da elaboração e revisão deste relatório.

Também não posso deixar de agradecer ao Marco Pereira, Álvaro Mateus e ao Jaime Correia, meus colegas estagiários, pelo apoio e disponibilidade ao longo deste percurso académico e pela sua paciência ao longo deste período.

Quero também agradecer a todos os meus amigos e colegas que me apoiaram neste percurso e que me ajudaram a conseguir alcançar os objetivos.

Por fim os mais importantes agradecimentos vão para a minha irmã pelo apoio, ajuda, motivação, incentivo e forças que me deu ao longo destes anos. Igualmente, deixo um especial agradecimento aos meus pais pela oportunidade, pelo apoio e pelo esforço que me permitiu apostar na minha formação para ser uma pessoa melhor. Também sem deixar de parte agradeço à restante família pelo suporte dado ao longo deste percurso. OBRIGADO.

Celso Rafael Clara Mendes

Conteúdo

1	Introdução	1
1.1	Contexto	1
1.1.1	A <i>MedicineOne</i>	2
1.1.2	Equipa	2
1.2	Objetivos	2
1.2.1	Visualização temporal	2
1.2.2	Deteção de eventos futuros no processo clínico	3
1.3	Motivação	4
1.4	Estrutura do documento	4
2	Estado da Arte	7
2.1	Visualização gráfica de informação	7
2.1.1	Atualmente no <i>MedicineOne 8</i>	8
2.1.2	A linha do tempo (<i>Timeline</i>)	9
2.2	Deteção de eventos futuros no processo clínico	9
2.3	Soluções existentes	10
2.3.1	<i>Lifelines</i>	10
2.3.2	<i>CLEF</i>	12
2.3.3	<i>KNAVE-II</i>	13
2.3.4	<i>TimeLine</i>	14
2.3.5	<i>AsbruView</i>	15
2.3.6	<i>SystmOne - Patient Timeline</i>	16
2.4	Conclusões	17
3	Gestão de Projeto	21
3.1	Metodologia de trabalho	21
3.2	Planeamento	22
3.2.1	Primeiro semestre	22
3.2.2	Segundo semestre	26
3.3	Processos de engenharia	29
3.4	Gestão de riscos	30
3.4.1	Riscos	31
3.4.2	Mitigação de riscos	33
4	Análise de Requisitos	35
4.1	Atores do sistema	35
4.2	<i>User Stories</i>	35
4.3	Requisitos funcionais	36
4.3.1	Requisitos gerais ao sistema	36

4.3.2	Requisitos do Presente	37
4.3.3	Requisitos do Passado	37
4.3.4	Requisitos do Futuro	38
4.4	Prototipagem	38
4.4.1	<i>Mockups</i> iniciais	38
4.4.2	<i>Mockups</i> finais	38
4.5	Restrições	42
4.5.1	Restrições técnicas	42
4.5.2	Restrições de negócio	43
4.5.3	Restrições legais	43
5	Arquitetura	45
5.1	Arquitetura do sistema	45
5.1.1	<i>Birds Eye View</i>	45
5.1.2	Camadas e componentes do sistema	46
5.1.3	Arquitetura lógica	49
5.1.4	<i>Model View Presenter</i>	50
5.2	Estruturas de dados criadas	51
5.3	Tecnologias	53
5.3.1	Desenvolvimento da linha do tempo	53
5.3.2	Deteção de eventos futuros no processo clínico	54
6	Trabalho Desenvolvido	55
6.1	Suite de testes	56
6.1.1	<i>Silverlight & WPF Timeline Control</i>	56
6.1.2	<i>WPF Scheduler Timeline Control</i>	57
6.1.3	<i>Winforms Scheduler Control + Range Control</i>	58
6.1.4	Conclusões	60
6.2	Protótipo	61
6.2.1	Passado	61
6.2.2	Presente	62
6.2.3	Futuro	62
6.2.4	Configuração	63
6.3	Deteção de eventos futuros no processo clínico	63
6.3.1	Características	63
6.3.2	<i>Missing Values</i>	67
6.3.3	<i>Data Sampling</i>	71
6.3.4	Testes e afinação do modelo	73
6.3.5	Classificação	77
6.3.6	Conclusões	78
7	Testes de Aceitação	81
8	Conclusões Finais e Trabalho Futuro	85
8.1	Conclusões finais	85
8.2	Trabalho futuro	86

A	<i>Mockups</i> Iniciais	91
A.1	Presente	91
A.2	Futuro	91
A.3	Passado	92
B	Requisitos Funcionais	93
B.1	Gerais ao Sistema	93
B.2	Presente	96
B.3	Passado	98
B.4	Futuro	100
C	Ordem de Implementação do Requisitos	103
D	Estimativas	105
E	Fluxo de Treino do Modelo	107
F	Classificador Webservice API	109
F.1	Pedido	109
F.2	Resposta	109

Lista de Figuras

2.1	Visualização do resumo clínico no <i>MedicineOne 8</i>	8
2.2	Interface do sistema <i>Lifelines</i>	11
2.3	Interface do sistema <i>Lifelines 3.0</i>	12
2.4	Interface do sistema <i>KNAVE-II</i>	14
2.5	Interface do sistema <i>Timeline</i>	15
2.6	Interface do sistema <i>AsbruView</i>	16
2.7	Interface do sistema <i>SystemOne Patient Timeline</i>	17
3.1	<i>Feature-Driven Development</i>	22
3.2	Lista de tarefas planeadas para o primeiro semestre	23
3.3	Diagrama de <i>Gantt</i> inicial relativo ao primeiro semestre	24
3.4	Diagrama de <i>Gantt</i> final relativo ao primeiro semestre	25
3.5	Lista de tarefas planeadas para o segundo semestre	26
3.6	Diagrama de <i>Gantt</i> referente ao segundo semestre	28
4.1	<i>Mockup</i> final para o presente	39
4.2	<i>Mockup</i> final para o passado	39
4.3	<i>Mockup</i> final para o futuro	40
4.4	Processar atividades futuras	40
4.5	Barra de ferramentas	40
4.6	<i>Overview</i> da linha do tempo	41
4.7	Listagem dos marcadores	41
4.8	<i>Tooltip</i>	41
4.9	Janela de configuração	42
5.1	<i>Birds Eye View</i>	46
5.2	Diagrama de camadas e componentes	47
5.3	Diagrama de componentes a desenvolver no sistema	48
5.4	Diagrama de <i>deployment</i> do serviço de classificação.	49
5.5	Arquitetura lógica	50
5.6	<i>Model View Presenter</i>	51
5.7	DTO relativos ao Futuro	51
5.8	DTO relativos ao Presente	52
5.9	DTO relativos ao Passado	52
6.1	<i>SIMILE WPF Timeline Control</i>	57
6.2	<i>WPF Infragistics Timeline Control</i>	57
6.3	<i>DevExpress Winforms Scheduler Control - Timeline View</i>	58
6.4	<i>DevExpress Winforms Scheduler Control - Gantt View</i>	59
6.5	Ecrã final relativamente ao momento do Passado.	61

6.6	Ecrã final relativamente ao momento do Presente.	62
6.7	Ecrã final relativamente ao momento do Futuro.	62
6.8	Ecrã final do menu de configuração.	63
6.9	Distribuição de idades por classe.	65
6.10	Frequência relativa da distribuição de género pelas classes.	66
6.11	Distribuição dos registos por classes.	66
6.12	Resultado do tratamento dos <i>missing values</i> recorrendo à média.	67
6.13	Resultado do tratamento dos <i>missing values</i> recorrendo à mediana.	68
6.14	Resultado do tratamento dos <i>missing values</i> recorrendo a uma regressão linear.	68
6.15	Resultado do tratamento dos <i>missing values</i> recorrendo a uma função quadrática.	69
6.16	Resultado do tratamento dos <i>missing values</i> recorrendo a uma função cúbica.	70
6.17	Resultado do tratamento dos <i>missing values</i> recorrendo a um polinómio de grau quatro.	70
6.18	Resultado do tratamento dos <i>missing values</i> recorrendo ao método KNN.	71
6.19	Primeiros quatro níveis da árvore de decisão.	79
A.1	<i>Mockup</i> inicial para o presente	91
A.2	<i>Mockup</i> inicial para o futuro	91
A.3	<i>Mockup</i> inicial para o passado	92
E.1	Componentes do fluxo da figura E.2.	107
E.2	Fluxo criado para o treino do modelo.	108

Lista de Tabelas

2.1	Comparação das diferentes soluções analisadas	19
2.2	Comparação do funcionamento dos diferentes sistemas analisados . .	20
3.1	Probabilidades de ocorrência dos riscos	30
3.2	Tabela de exposição dos riscos	33
6.1	Resultados das experiências realizadas aos componentes.	60
6.2	Métricas de performance do sistema	72
6.3	Instâncias identificadas corretamente	72
6.4	Métricas de performance do sistema.	75
6.5	Instâncias identificadas corretamente	76
6.6	Métricas de performance aquando da validação do modelo.	78
6.7	Instâncias identificadas corretamente aquando da validação do modelo.	78
7.1	Conjunto de testes de aceitação realizados à linha do tempo.	81
C.1	Listagem ordenada dos requisitos pela ordem a implementar	104
D.1	Estimativas realizadas para cada requisito	106

Lista de Acrónimos

EHR	<i>Electronic Health Record</i>
CLEF	<i>Clinical e-Science Framework</i>
KNAVE	<i>Knowledge-based Navigation of Abstractions for Visualization and Explanation</i>
FDD	<i>Feature-Driven Development</i>
TFS	<i>Team Foundation Server</i>
MVP	<i>Model View Presenter</i>
UI	<i>User Interface</i>
US	<i>User Stories</i>
WPF	<i>Windows Presentation Foundation</i>
SQL	<i>Structured Query Language</i>
WCF	<i>Windows Communication Foundation</i>
SOAP	<i>Simple Object Access Protocol</i>
HTTP	<i>HyperText Transfer Protocol</i>
HTTPS	<i>HyperText Transfer Protocol Secure</i>
DTO	<i>Data Transfer Object</i>
KNN	<i>K-Nearest Neighbor</i>
JSON	<i>JavaScript Object Notation</i>
REST	<i>Representational State Transfer</i>
AUC	<i>Area Under Curve</i>
ACC	<i>Accuracy</i>
ROC	<i>Receiver Operating Characteristic</i>

Capítulo 1

Introdução

Este relatório de estágio é referente à unidade curricular Dissertação/Estágio, do Mestrado em Engenharia Informática, da área de engenharia de *software*, da Faculdade de Ciências e Tecnologia da Universidade de Coimbra. O estágio está a ser realizado na *Medicineone, Life Sciences Computing S.A.*, em Coimbra, com a orientação do Professor Doutor César Teixeira, docente no Departamento de Engenharia Informática da faculdade acima referida e pelo Engenheiro Nuno Teixeira, na *MedicineOne*.

Neste capítulo é descrito o contexto do estágio, os seus objetivos, assim como a restante estrutura do presente documento.

1.1 Contexto

Na área da saúde a quantidade de informação que existe, relativa a cada utente, pode ter um volume considerável, pois é reunida desde que nascemos, ou por vezes ainda antes, durante a gestação. Com este crescimento e em conjunto com a evolução da tecnologia ao longo dos anos foi possível transformar estes processos, em formato de papel, que guardam a informação individual de cada utente para um processo clínico eletrónico (*Electronic Health Record (EHR)*). Com esta evolução para EHR, surgiu a necessidade de permitir que o utilizador, o profissional clínico, possa com alguma facilidade observar e estudar o processo para realizar o respetivo diagnóstico do estado de saúde do utente. Para além de permitir uma melhor gestão do processo individual do utente, os EHR também facilitam a partilha de informação clínica de cada utente entre as diversas unidades de saúde, permitindo que os diagnósticos sejam mais precisos e realizados mais rapidamente, pois assim é possível num curto intervalo de tempo verificar dados do utente que tenham sido registados noutras unidades de saúde, possibilitando a oferta ao utente de um melhor cuidado de saúde.

Atualmente esta informação encontra-se disponível em diversos formatos (relatórios, tabelas, etc.), espalhada por vários itens no processo clínico do utente. O profissional de saúde ao consultar o processo tem de aceder aos vários itens para consultar toda a informação de que necessita, isto é um processo de consulta demorado não permitindo que o profissional possa ter uma noção do estado do utente de forma simples e rápida.

Com a utilização dos sistemas EHR torna-se possível a utilização de novas técnicas para a visualização dos mesmos, como é o caso de estudo neste projeto, que passa pela visualização do processo clínico com recurso a uma linha do tempo.

1.1.1 A *MedicineOne*

A *MedicineOne*¹ é uma empresa de desenvolvimento de *software* para o mercado da saúde, tendo como principal foco o desenvolvimento de software para a gestão de pequenas a grandes unidades de saúde, sistemas de saúde públicos e privados. A empresa tem um âmbito de ação global pois é efetuada a gestão, de uma forma integrada num único sistema, da área clínica, administrativa, logística, de gestão, financeira e estatística.

1.1.2 Equipa

O estagiário foi integrado numa das equipas de desenvolvimento da *MedicineOne*, sendo que esta equipa é supervisionada pelo orientador do mesmo. Esta equipa é composta por seis elementos, inclusive o estagiário, sendo que o seu foco de trabalho é o desenvolvimento de novas *features*, suporte e manutenção de alguns dos módulos da aplicação, pelos quais é responsável.

1.2 Objetivos

Este projeto de estágio encontra-se dividido em duas fases, numa primeira fase pretende-se que seja efetuado o estudo e a implementação de um mecanismo de visualização e análise temporal do processo clínico, em que com este mecanismo se pretende uma nova forma de visualização do processo clínico. Na segunda fase deste estágio pretendeu-se a realização de um estudo para a criação de um meio complementar de diagnóstico que permita identificar possíveis relacionamentos entre os problemas de saúde e o processo clínico e identificar a possibilidade do futuro aparecimento de um problema de saúde.

1.2.1 Visualização temporal

Para a primeira fase deste estágio foi apresentada a proposta do estudo e da implementação de um protótipo de um mecanismo para visualização temporal de informação clínica de um utente. Este mecanismo consiste em mostrar a informação do utente com recurso ao posicionamento da mesma numa linha do tempo, que permitirá ao utilizador consultar facilmente a informação e rapidamente verificar o estado de saúde do utente. Para além da visualização da informação também é possível realizar as mais diversas tarefas sobre os dados apresentados.

Este mecanismo será disponibilizado como uma nova funcionalidade no *MedicineOne 8*, que é uma solução disponibilizada pela empresa para a gestão de unidades de saúde. O módulo a desenvolver será integrado na área de consulta do processo clínico do utente, e será uma alternativa para a consulta da informação. Atualmente a consulta dessa informação é efetuada com recurso a tabelas, imagens, relatórios, etc.

A implementação desta funcionalidade passa pela representação da informação clínica através de uma linha do tempo dividida em três épocas: o passado, o presente e o futuro.

¹<http://www.medicineone.net/>

- **Passado:** Neste momento pretende-se mostrar o historial clínico do utente até à data atual. Aqui o utilizador também poderá observar graficamente relações entre eventos, por exemplo: Uma interação entre medicamentos pode aumentar o risco de uma determinada patologia, se após essa interação a patologia se verificar o utilizador poderá verificar essa relação na linha do tempo.
- **Presente:** Nesta secção pretende-se representar a informação clínica atual do utente, isto é representar a informação mais importante do historial clínico do utente desde uma data recente, sendo o tamanho do intervalo predefinido pelo utilizador. Esta informação contém apenas a entrada, mais recente, relativa a cada grupo de informação que tenha realizado desde data inicial do intervalo até à data atual. Neste momento o utilizador pode facilmente compreender se uma entrada está com um valor se encontra dentro, acima ou abaixo do intervalo considerado normal para essa métrica.
- **Futuro:** No futuro é mostrada ao utilizador informação relativa a eventos futuros que já se encontrem ou possam ser agendados para realização por parte do utente, por exemplo consultas e exames médicos.

A linha do tempo deverá permitir ao utilizador facilmente perceber o estado de saúde do utente, necessidades deste em termo de prescrições de medicamentos, exames e consultas de especialidade. O profissional clínico também deverá ser capaz de interagir com a linha do tempo permitindo visualizar em mais detalhe cada evento, navegar ao longo da linha do tempo ou até mesmo realizar operações de *zoom* para uma melhor visualização/compreensão de um conjunto de eventos num intervalo temporal. Para além das operações relacionadas com a visualização de informação, a linha do tempo também deverá permitir ao utilizador realizar outras tarefas relacionadas com o processo do utente, como por exemplo renovar o receituário, agendar exames, marcar consultas, entre outras.

1.2.2 Deteção de eventos futuros no processo clínico

Numa segunda fase deste projeto foi efetuado o estudo para a implementação de um sistema que possa complementar a informação da linha do tempo, com a identificação de possíveis problemas de saúde que um determinado utente possa vir a contrair.

Este módulo deverá, através dos dados presentes no processo clínico, dizer a possibilidade do utente vir a contrair a patologia em análise.

Para a esta fase inicial deste módulo e tendo em conta o tempo disponível para a realização do mesmo é pedido que o sistema faça a análise dos dados clínicos do utente e identifique se este possa vir a contrair *diabetes tipo II*. Foi escolhida esta patologia uma vez que este é um problema que tem um grande impacto na sociedade e que um utente pode vir a contrair devido ao seu estilo de vida (tipo de alimentação que faz, hábitos de saúde, entre outros).

O sistema deverá mostrar ao utilizador o resultado da análise e mostrar graficamente o relacionamento desse resultado com os itens presentes no processo clínico, que se encontrem representados na linha temporal. Esses resultados deverão ser mostrados com recurso à representação de relações entre os vários eventos presentes na linha do tempo.

1.3 Motivação

A visualização de um processo clínico pode ser uma tarefa que consome algum tempo uma vez que o volume de informação contida no processo pode ser elevada. Isto leva à necessidade de ferramentas para a visualização da informação que facilitem a tarefa do profissional clínico.

Conhecendo este facto começou-se a idealizar a implementação de uma nova forma de visualizar a informação que se encontra no processo clínico do utente. Sendo que este novo método de visualização permite que o profissional clínico visualize essa mesma informação para tomar as respetivas decisões num menor intervalo de tempo e com a mesma exatidão como se estivesse a consultar o processo clínico normalmente.

Esta ideia surgiu do Engenheiro João Miguel, fundador da *MedicineOne*, em que este a estruturou e a apresentou ao aluno no início do estágio. Já no decorrer do estágio verificou-se que para além do Engenheiro João Miguel alguns "*clientes da empresa*"/**médicos/Profissionais Clínicos**, também tinham tido uma ideia semelhante e em conversas entre os mesmos e o Engenheiro João Miguel verificaram que existia a necessidade da implementação de uma linha do tempo para a visualização do processo clínico do utente.

1.4 Estrutura do documento

O presente documento tem como objetivo documentar o trabalho realizado ao longo do decorrer do estágio e encontra-se estruturado como descrito a seguir.

Capítulo 2 - Estado da Arte - Neste capítulo é apresentado o estudo do estado da arte, a análise das soluções que foram encontradas e de como é realizada, atualmente, a visualização do processo clínico no *MedicineOne 8*.

Capítulo 3 - Gestão de Projeto - Neste terceiro capítulo é descrita a metodologia utilizada para o desenvolvimento do projeto, o planeamento realizado para os dois semestres, assim como os processos de engenharia utilizados e os riscos inerentes ao projeto.

Capítulo 4 - Análise de Requisitos - Este capítulo contém a análise de requisitos efetuada, com a identificação dos atores do sistema, a definição das *User Stories*, os *mockups* finais, a enumeração dos requisitos identificados e as restrições inerentes ao desenvolvimento do projeto.

Capítulo 5 - Arquitetura - No quinto capítulo é apresentada a arquitetura do sistema atual do *MedicineOne 8* e explicada a arquitetura do componente a desenvolver, assim como as estruturas de dados que serão necessárias para dar suporte à linha do tempo.

Capítulo 6 - Trabalho Desenvolvido - Neste capítulo é apresentado todo o trabalho realizado no decorrer do estágio, assim como os resultados obtidos nas experiências realizadas relativamente às duas partes do estágio. Também neste capítulo são apresentados os ecrãs finais da linha do tempo.

Capítulo 7 - Testes de Aceitação - Neste capítulo são apresentados os testes de aceitação para o projeto e os respetivos resultados.

Capítulo 8 - Conclusões Finais e Trabalho Futuro - Neste capítulo é apresentada uma conclusão sobre o decorrer do estágio, sendo realizado um apanhado dos aspetos que bem sucedidos e as principais dificuldades enfrentadas. Para além das conclusões também são apresentadas algumas sugestões que ficam para trabalho a realizar futuramente.

Capítulo 2

Estado da Arte

Neste capítulo pretendo contextualizar o leitor sobre a temática de visualização de informação, bem como descrever como atualmente um médico visualiza o processo clínico do utente no sistema em que será integrado o componente.

Para além de como é visualizada a informação atualmente também é apresentado ao leitor o que atualmente existe acerca da utilização de uma linha do tempo em sistemas que permitam ajudar o utilizador a visualizar grandes quantidades de informação e a tomar decisões com base nessa informação. Por vezes essa informação encontra-se disponível em quantidades consideráveis que obrigam a que o utilizador dispense algum tempo para a análise dessa informação e posteriormente tome as respetivas decisões.

2.1 Visualização gráfica de informação [1, 2]

Com o evoluir da tecnologia a quantidade de informação disponível para os profissionais clínicos analisarem tem vindo a aumentar em termos de quantidade e de complexidade. O que nos leva a um dilema: "Como é que se pode visualizar a informação clínica disponível de forma fácil e rápida?" Isto passa a ser possível com recurso a algumas técnicas de visualização de informação que podem ser aplicadas com recurso à tecnologia. Estas técnicas passam pela representação da informação em 1D, 2D, 3D, multidimensional, em árvore, em rede ou com recurso a mecanismo de representação de informação temporal. Para além da visualização da informação, a realização de determinadas operações sobre a mesma pode facilitar a sua análise e compreensão. As operações que nos permitem uma melhor análise dos dados são: a representação de um *overview* dos mesmos, a realização de *zoom*, filtrar os dados que se pretendem visualizar, identificar possíveis relações, analisar em mais detalhe conforme a necessidade, manter o histórico das operações realizadas e permitir a extração de *subsets* de dados para uma melhor análise. Com isto o utilizador pode analisar mais eficazmente a informação que tem ao seu dispor e rapidamente chegar às devidas conclusões.

No caso deste projeto pretende-se a representação gráfica da informação temporal, sendo que tendo em conta o tipo da informação a representar, será utilizada uma linha do tempo.

2.1.1 Atualmente no *MedicineOne 8*

O *MedicineOne 8* é a principal solução da empresa, sendo que é nesta aplicação que será integrado o componente desenvolvido no decorrer do estágio. Esta solução permite que as diferentes organizações de saúde possam efetuar toda a sua gestão, desde o cuidado do utente até a outras áreas como por exemplo a área financeira da organização.

Como o mecanismo a desenvolver tem o objetivo de estar integrado no *software MedicineOne 8*, de seguida será apresentado como atualmente é visualizada a informação presente no processo clínico de um utente.

The screenshot displays the MedicineOne 8 interface for a patient named António Ferreira. The patient's details include: Nº processo: 138, Sexo: Masculino, Nº nacional: Sem informação, Nº trabalhador: Sem informação, Data nasc: 28/10/1978, and Próxima consulta: 11/.

The interface shows two tables of clinical data:

Últimos formulários registados	
Data	Nome
06/08/2015	testar_formulario
13/08/2015	Complemento FAM
03/12/2015	Complemento FAM
15/12/2015	Cardiologia teste
16/12/2015	Complemento FAM

Últimos atos clínicos realizados		
Data	Código	Nome
13/01/2016	16.05	*Biópsias urológicas
13/01/2016	16.05	*Biópsias urológicas
12/01/2016	16.05	*Biópsias urológicas
12/01/2016	16.05	*Biópsias urológicas

Below the tables, there is a 'Resultados ...' button and a 'Família' section with a table header: Fotografia, Nº processo, Nome, Parentesco, Idade.

Figura 2.1: Visualização do resumo clínico no *MedicineOne 8*

Como é possível observar na figura 2.1, a visualização da informação clínica é efetuada com recurso a tabelas estruturadas onde aparecem as datas de realização e uma referência ao método complementar de diagnóstico. O utilizador para consultar mais informação relativa a esse método complementar de diagnóstico deverá consultar os detalhes do mesmo. Este método de visualização não é o mais favorável para analisar múltiplos eventos ocorridos ao longo do tempo, uma vez que o utilizador apenas tem a informação em tabelas estando esta ordenada por datas.

Podemos observar que o utilizador para ver mais informação clínica do utente tem de efetuar *scroll* na página para assim conseguir ver a restante informação que se encontra mais abaixo.

2.1.2 A linha do tempo (*Timeline*)

As linhas do tempo permitem-nos a visualização de eventos por ordem cronológica e podem ser encontradas ao longo das nossas tarefas do dia-a-dia, por exemplo, na programação da nossa televisão, na nossa agenda e nas redes sociais onde as diferentes publicações são mostradas ao longo de uma linha do tempo. Para além destes exemplos do nosso dia-a-dia as linhas do tempo também são utilizadas para mostrar o desenrolar de eventos históricos, de eventos naturais (por exemplo a evolução de uma pandemia) e também são utilizadas na gestão de projetos (diagramas de *Gantt*).

Para a representação dos dados numa linha do tempo podem-se utilizar diversos itens gráficos para que se possam representar os diferentes tipos de eventos, categorias, importância e duração. Por exemplo para representar os eventos pode-se recorrer a pontos, para a sua duração utilizar linhas/retângulos, para representar a categoria de cada evento pode-se utilizar um pré-determinado código de cores e para identificar eventos mais relevantes, que necessitem de mais atenção por parte do utilizador, pode-se recorrer a uma seleção do mesmo.

2.2 Detecção de eventos futuros no processo clínico

Com a elevada quantidade de informação que os sistemas EHR têm disponíveis, existe a possibilidade de utilizar essa informação em sistemas de reconhecimento de padrões. Estes sistemas poderão analisar esses dados e detetar possíveis padrões que o médico não venha a detetar ou até mesmo antes deste os detetar. Mesmo com esta quantidade de informação muitos dos sistemas EHR não possibilitam a sua utilização por questões legais e éticas.

As diferentes entidades que fazem a gestão dos sistemas EHR acabam por estar a desperdiçar algumas oportunidades na medida em que não tiram proveito dos dados que têm ao seu dispor. Uma vez que para além destas beneficiarem, os próprios utentes também teria algum benefício na medida em que poderiam tentar evitar algumas patologias que podem estar relacionadas com os hábitos de vida.

Atualmente já existem alguns estudos que nos falam sobre a aplicação de técnicas de reconhecimento de padrões para a previsão e deteção de doenças. Podemos ver alguns exemplos, como é o caso apresentado por *Gandhi* [3], onde se pretende aplicar metodologias de *machine learning* para a previsão de diabetes, ou até mesmo como nos é mostrado por *Plis* [4], onde são utilizadas técnicas similares para o acompanhamento e previsão dos níveis de glucose no sangue de doentes diabéticos. Não é só a patologia diabetes que merece atenção suficiente para a utilização de técnicas de reconhecimento de padrões, também existem estudos na utilização das mesmas técnicas para a previsão e deteção de cancro [5], doença de *Parkinson* [6], entre outras. Focando no objetivo deste trabalho pretendemos utilizar técnicas de reconhecimento de padrões para a deteção de diabetes, fazendo uso dos dados que temos disponíveis no processo clínico e disponibilizando ao utilizador informação sobre os vários aspetos que levaram o sistema a tomar a respetiva decisão.

Com a aplicação de técnicas de reconhecimento de padrões, existe aqui uma grande variedade de ferramentas que podem ser criadas para ajudar o profissional clínico, principalmente os médicos, nas suas análises dos processos clínicos. Com estas ferramentas pode ser possível detetar problemas logo aquando do seu surgimento ou mesmo antes, de forma a que estes ainda não estejam numa fase mais avançada que a altura em que o tratamento, do mesmo, ainda é viável. Com isto, pretende-se a prevenção e deteção precoce dos problemas para que o utente possa ter os melhores cuidados de saúde.

2.3 Soluções existentes

Atualmente já existem algumas soluções que recorrem a uma linha do tempo para a visualização de informação. Algumas das soluções que existem são da área da saúde, as quais estão mais próximas do objetivo de estudo deste trabalho, existindo também outras soluções que são de áreas diferentes. De seguida, são apresentadas algumas dessas soluções.

Na área da saúde existem vários sistemas para a visualização de dados com recurso a uma linha do tempo, isto deve-se ao facto de os dados nesta área serem dependentes do tempo, pois é necessário ter conhecimento de quando são recolhidos para a sua análise. Entre os vários sistemas que recorrem a uma linha do tempo, existem alguns que estão mais virados para o cuidado do utente e outros que são dedicados à investigação que utilizam dados de vários utentes de forma a ajudar os investigadores nos projetos.

De seguida vamos apresentar alguns desses sistemas da área da saúde e explicar um pouco cada um destes.

2.3.1 *Lifelines* [7–10]

O *Lifelines* apareceu em 1996 como resultado num estudo onde se pretendia utilizar uma linha do tempo para a visualização do histórico pessoal e tinha como principais áreas de destino a área clínica e também a área judicial, apesar de também ser possível utilizar com dados referentes a outras áreas.

Neste sistema, o utilizador tem a possibilidade de visualizar os dados clínicos ao longo de uma linha do tempo em que estes se encontram divididos por categoria e tipo de dados.

Esta solução foi desenvolvida para profissionais clínicos, tendo como objetivo o cuidado do utente e também para os profissionais judiciais, sendo que serviu principalmente para a consulta dos processos judiciais. A gravidade/relevância de cada evento é representada com recurso a diferentes espessuras das linhas onde, por exemplo, conforme o estado clínico de um utente vai melhorando, a linha fica mais fina e caso este pior a linha fica mais larga. Alguns dos dados são transformados para escalas quantitativas, por exemplo para os valores da pressão arterial de um utente em vez de se mostrar o valor medido é mostrado ao médico se este valor está num intervalo considerado normal, está elevado ou baixo, com estas categorizações consegue-se simplificar a interpretação dos dados. Também se recorre ao uso de diferentes cores ao longo da linha do tempo para identificar a severidade de um evento e relações entre diferentes eventos. Por exemplo o diagnóstico de um problema de saúde terá

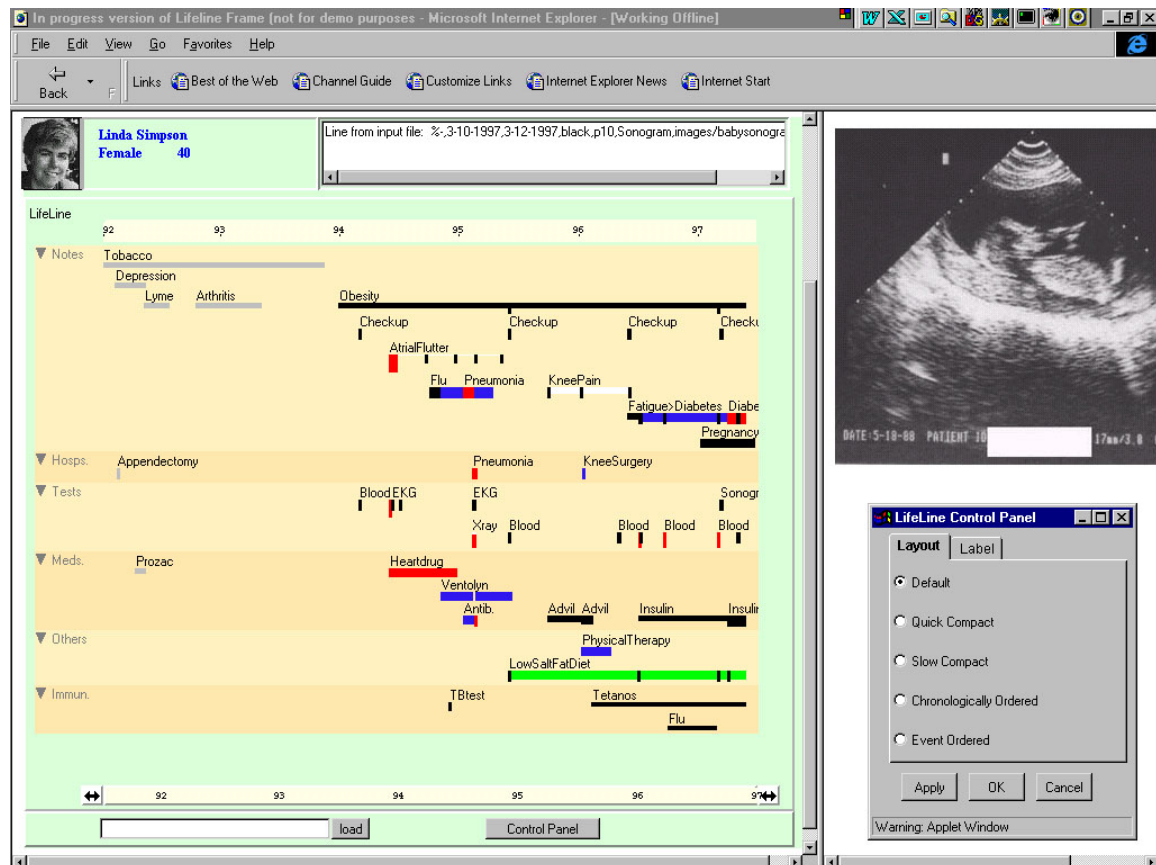


Figura 2.2: Interface do sistema *Lifelines*, onde se pode visualizar a linha do tempo no lado esquerdo do ecrã. (Imagem retirada de "<http://www.cs.umd.edu/hcil/lifelines/>")

a mesma cor que os exames realizados derivados desse diagnóstico, assim como os tratamentos prescritos para esse diagnóstico.

Para além das características acima apresentadas esta solução também permite ao utilizador interagir com a linha do tempo para uma mais fácil interpretação dos dados. Na linha do tempo apenas é mostrado o evento sobre a forma de ponto/linha, conforme a duração do mesmo, em conjunto com um título, caso o utilizador pretenda obter mais informação sobre o mesmo tem a possibilidade de clicar nele e observar a informação mais detalhada do evento. Para além disto o utilizador pode também realizar operações de *zoom* para ter uma melhor visão sobre os eventos ao longo do tempo. O utilizador também pode efetuar pesquisas na linha do tempo, sendo que os eventos em que exista alguma relação com a pesquisa são selecionados para que o utilizador os possa facilmente identificar, por exemplo, procurando por uma patologia sempre que um evento refira essa patologia esse evento é selecionado.

Lifelines 3.0 [11]

Do projeto *Lifelines* surgiu mais tarde uma nova versão, o *Lifelines 3.0*, que consistiu na adaptação e melhoramento da versão original, tendo como destino uma área diferente que a da saúde. Este sistema tem como principal objetivo a gestão dos processos militares individuais nos Estados Unidos da América.

Com esta versão preparada para os dados militares foi possível também corrigir

alguns problemas que foram surgindo nas versões anteriores que eram focadas na área da saúde aquando a sua utilização com os dados militares.

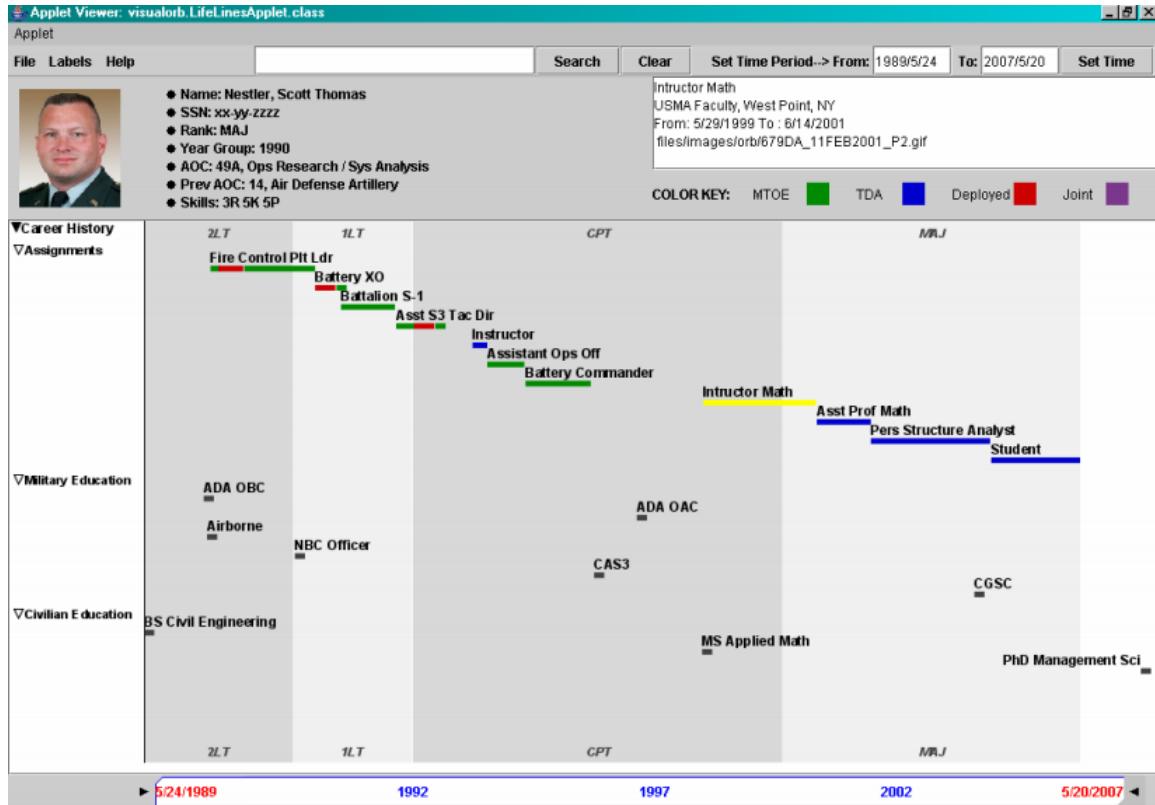


Figura 2.3: Interface do sistema *Lifelines 3.0* (Imagem retirada do artigo *SystemOne Patient Timeline* [11].)

Esta linha do tempo é simples uma vez que a informação representada é referente a eventos ocorridos no decorrer de uma carreira militar. Os dados encontram-se categorizados de diversas formas, como se pode observar na figura 2.3, em primeiro lugar os dados são divididos pelas categorias em que o oficial já passou ao longo da carreira, em segundo lugar os dados são codificados por um código de cores conforme os diferentes tipos de tarefas que este realizou e por fim os eventos podem ser agrupados pela sua natureza (por exemplo: educação militar, educação civil, etc.)

Neste sistema o utilizador ao selecionar um evento pode visualizar a informação mais detalhada numa caixa de texto que aparece no canto superior direito do ecrã, para além desta funcionalidade também é possível pesquisar eventos que possam existir.

2.3.2 CLEF [7, 12]

O *Clinical e-Science Framework* apresentado por Hallet [12], o sistema CLEF é composto por três linhas do tempo paralelas em que cada uma representa a informação relativa a diagnósticos, tratamentos e a exames realizados. Este sistema tem como principal utilizador o profissional clínico e destina-se à interação deste com o utente visando o seu bem-estar.

Neste sistema o utilizador pode facilmente visualizar os dados do paciente ao longo da linha do tempo e, para além desta visualização, também é oferecido ao utilizador

um pequeno sumário a complementar a informação representada, permitindo assim uma rápida conclusão sobre o estado de saúde do utente.

O utilizador neste sistema pode facilmente visualizar eventos relacionados, pois as relações entre estes podem ser identificadas pelo sistema e mostradas ao utilizador. Existem três tipos de relações que são identificadas pelo sistema, sendo estas relações retóricas (causa-efeito), relações ontológicas e relações atributivas (relacionam uma intervenção/problema a um específico local no corpo por exemplo um órgão).

O sistema recorre a linguagem natural para a criação dos relatórios que ajudam a interpretar os eventos da linha do tempo e os relatórios textuais que indicam o estado de saúde do utente. Estes relatórios podem ser gerados a partir de um determinado conjunto de eventos pré-selecionados pelo utilizador ou podem ser gerados para oferecer suporte à linha do tempo de onde o utilizador pode arrastar eventos para melhor visualizar e interpretar as suas relações.

2.3.3 *KNAVE-II* [7, 13–15]

KNAVE-II é um sistema de visualização e exploração de dados temporais em diferentes níveis de abstração temporal. Este sistema à semelhança dos apresentados anteriormente é utilizado pelos profissionais clínicos visando o cuidado do utente.

Neste sistema o utilizador pode ver os dados do utente separados em múltiplas linhas do tempo conforme estas são selecionadas no menu que se encontra no lado esquerdo da interface, como é possível observar na figura 2.4. Neste menu encontram-se diferentes ontologias referentes aos dados recolhidos e guardados na base de dados ou a resultados da aplicação de processos de abstração temporal, para uma mais fácil interpretação da informação.

O utilizador tem a possibilidade de interagir com os dados, para uma melhor visualização e perceção, alterando a granularidade dos mesmos. Para além desta interação, o sistema também permite que as linhas do tempo sejam absolutas ou relativas, para ser possível visualizar os dados sem tratamento ou após a aplicação de abstrações à informação contida nos mesmos, estas abstrações dos dados podem passar pela divisão dos valores em várias classes (por exemplo a classificação dos valores como baixo, normal ou alto).

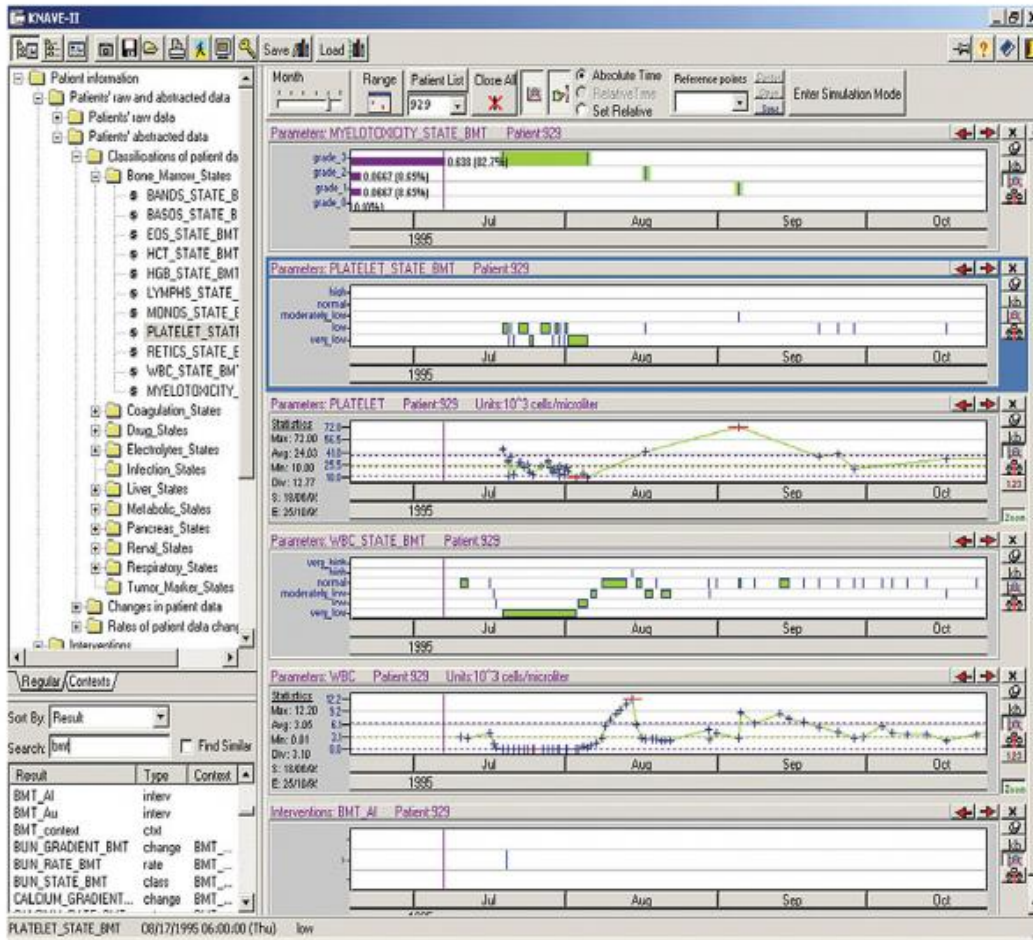


Figura 2.4: Interface do sistema *KNAVE-II*, onde se pode observar do lado direito as diferentes linhas do tempo e do lado esquerdo o menu com as possíveis ontologias que o utilizador pode escolher (Imagem retirada de "<http://biomedicalcomputationreview.org/content/clinical-decision-support-providing-quality-healthcare-help-computer>".)

2.3.4 *TimeLine* [7, 16]

O sistema *Timeline*, apresentado em *Bui et al.* [16], mostra uma linha do tempo em que o profissional clínico pode facilmente consultar o processo do utente e até mesmo visualizar exames na mesma, como é possível observar na figura 2.5, em que na linha do tempo se encontram alguns resultados de exames, sendo neste caso de um paciente neuro-oncológico.

Os dados do utente são organizados numa árvore para a sua visualização em linhas do tempo separadas. O utilizador neste sistema para visualizar a informação pretendida deverá expandir os pontos que pretende visualizar através do controlo que se encontra no lado esquerdo do ecrã, sendo que ao efetuar a expansão de um item a respetiva informação será mostrada na linha do tempo.

Nesta linha do tempo a informação do utente é mostrada conforme o utilizador do sistema pretende, sendo necessário este selecionar que tipo de informação deseja

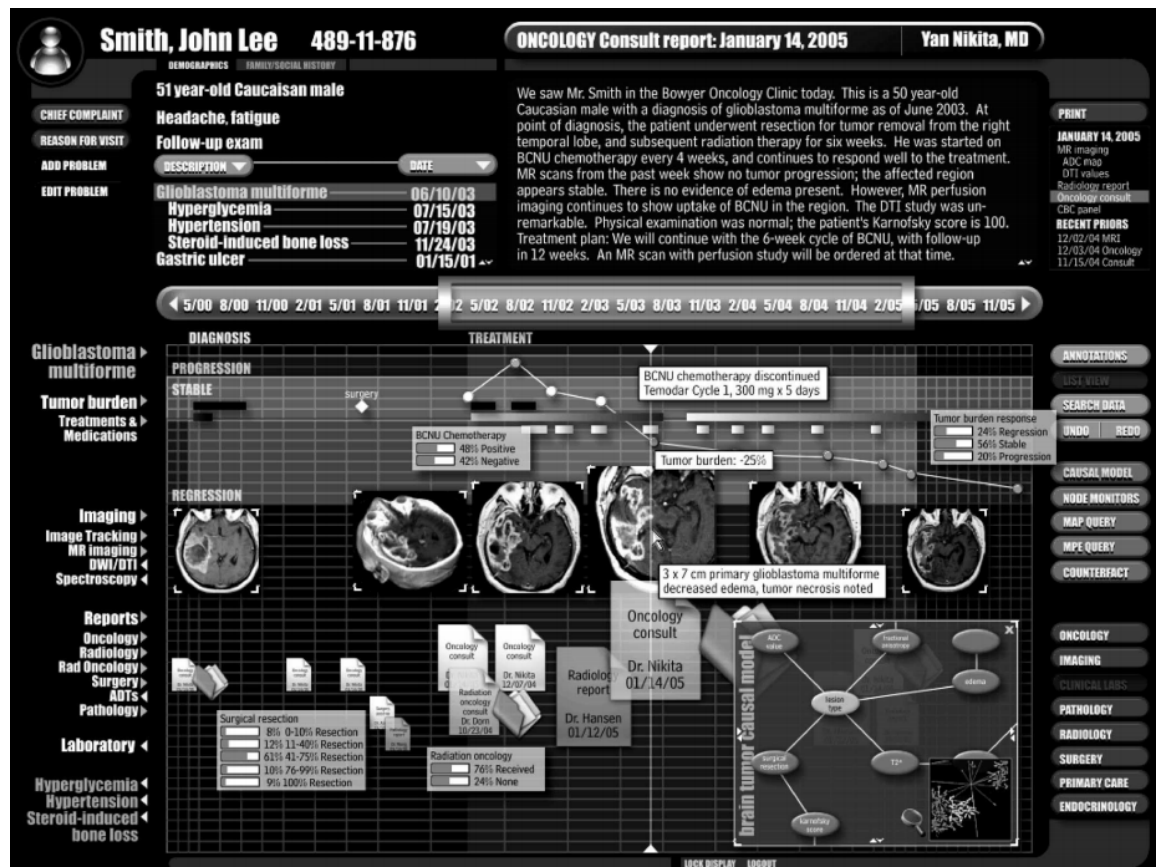


Figura 2.5: Interface do sistema *TimeLine*, é visível do lado esquerdo da imagem a árvore da informação. Na linha do tempo são visíveis alguns resultados de exames. (Imagem retirada do artigo [16].)

visualizar. Com isto não é possível que o utilizador possa rapidamente perceber o estado de saúde do utente sem que tenha a necessidade de andar a navegar na árvore que contém a informação disponível.

2.3.5 *AsbruView* [7, 17, 18]

O sistema *AsbruView* permite a representação de relações temporais e hierárquicas entre os planos de tratamentos, estas representações são realizadas utilizando uma perspetiva de visualização em 3D. Os planos de tratamentos podem ser alinhados ao longo do tempo de diferentes formas para que as diferentes relações entre estes possam ser facilmente interpretadas. Para um mais fácil entendimento dos objetos gráficos na linha do tempo, recorrem a metáforas de elementos do dia-a-dia do utilizador (por exemplo semáforos, faixas, etc. como é possível observar na zona superior da figura 2.6).

Para além desta perspetiva 3D, também é apresentada uma perspetiva 2D, linha do tempo, mais virada para os aspetos temporais dos planos de tratamento, onde o utilizador pode observar e compreender melhor os diferentes pontos de cada plano e quando estes deverão ocorrer. Esta linha do tempo é relativa, pois cada um dos eventos depende da sua relação temporal com outros eventos.

Como referido anteriormente este sistema é um sistema mais focado na aplicação de planos de tratamento ao utente e não tanto para a visualização do processo clínico

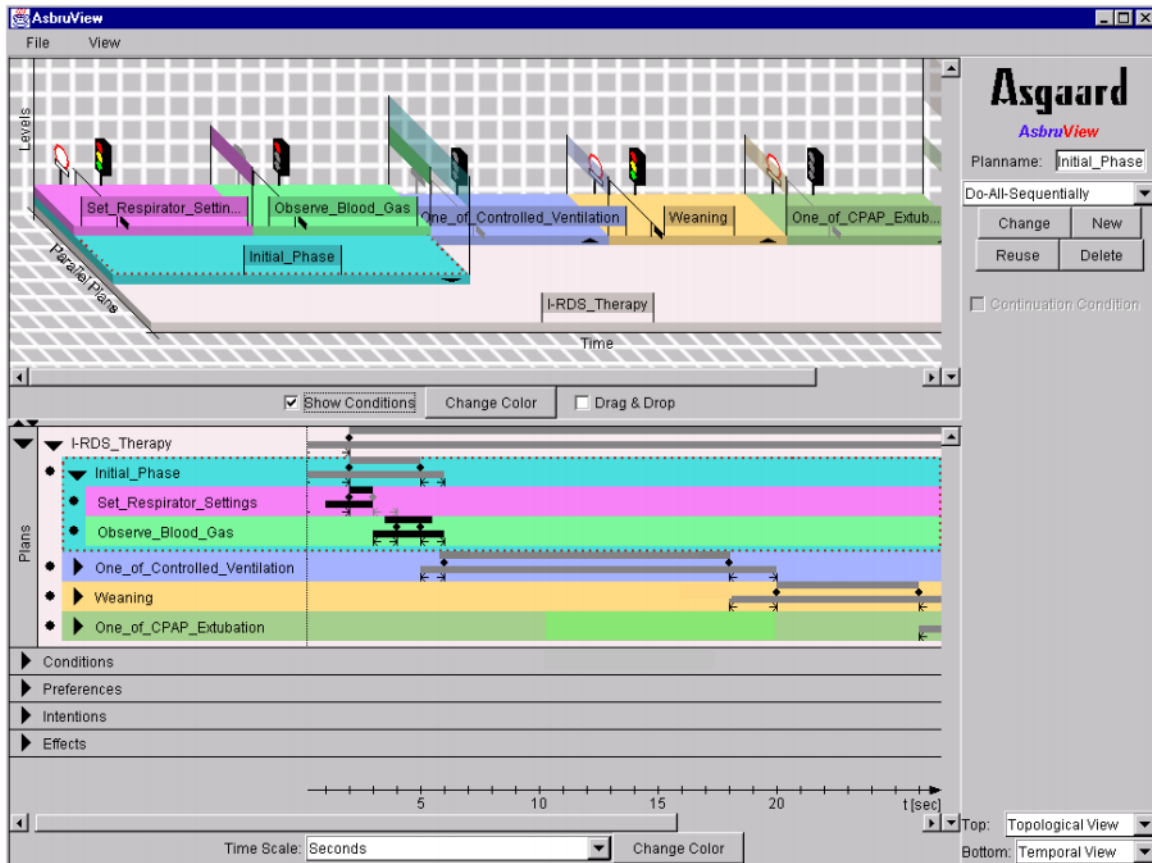


Figura 2.6: Interface do sistema *AsbruView*, onde na zona superior se pode observar uma vista topológica dos planos de tratamento numa perspetiva em 3D e na zona mais abaixo é possível observar em mais detalhe, através de uma linha do tempo relativa, o plano representado em cima. (Imagem retirada do artigo [18].)

do utente. O principal utilizador deste sistema é o profissional clínico, pois é este quem deve fazer o planeamento do tratamento que o utente vai realizar e definir as diferentes etapas desse tratamento.

2.3.6 *SystemOne - Patient Timeline* [19, 20]

O *SystemOne* é um sistema de gestão de registos de saúde individuais que se encontra em funcionamento nas unidades de saúde do Reino Unido. Este sistema para além da gestão da informação clínica de cada utente também permite a visualização da informação clínica dos utentes com recurso a uma linha do tempo.

Neste sistema o utilizador pode facilmente visualizar a informação do utente numa linha do tempo, sendo que esta informação é separada em três camadas, como é possível observar no topo da figura 2.7. A camada do topo, a verde, apresenta os problemas que o paciente teve ao longo do tempo, a segunda camada, a laranja, apresenta a informação relativa aos medicamentos que foram prescritos ao utente, e por fim a camada que se encontra a vermelho apresenta a informação clínica relevante do utente.

Na zona central do ecrã temos a linha do tempo com uma vista mais detalhada. Nesta zona a informação segue o esquema de cores explicado anteriormente, mas os dados mostrados são referentes a diferentes itens que o utilizador pode seleccionar, por

exemplo os valores da pressão arterial, medicamentos, etc.

No fundo da página é possível observar, com um mais informação, os diferentes eventos apresentados na linha do tempo, sendo que estes são destacados conforme o utilizador passa com o rato por cima dos mesmos na linha do tempo.

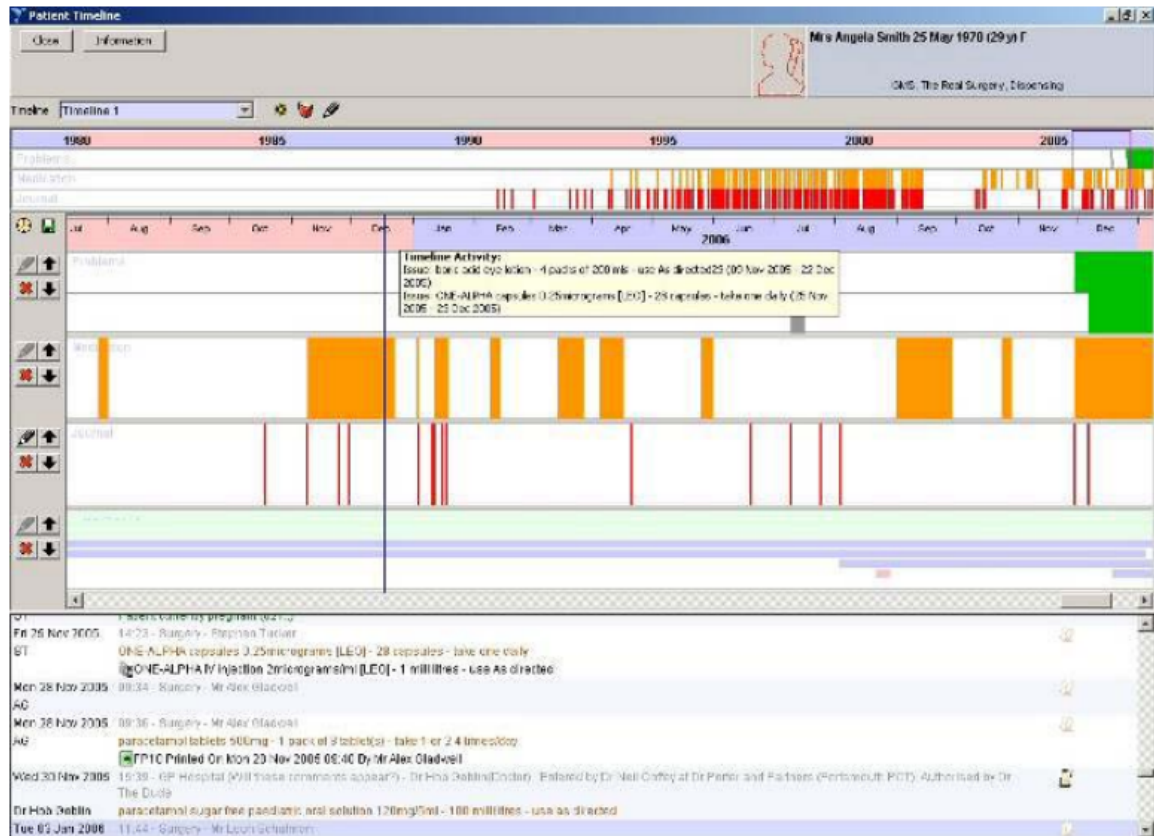


Figura 2.7: Interface do sistema *SystemOne Patient Timeline* onde é possível observar as três principais áreas do sistema: *overview* da linha do tempo, as linhas do tempo relativas a cada item e no fundo vista em detalhe de eventos. (Imagem retirada do manual do utilizador do *SystemOne Patient Timeline* [20].)

2.4 Conclusões

Este projeto tem como objetivo a criação de uma linha do tempo que permitirá ao utilizador observar, com relativa simplicidade, o estado de saúde do utente e devendo mostrar a informação do processo clínico do mesmo. Podemos concluir, após efetuada a análise das soluções apresentadas acima, que existem algumas soluções que se encontram próximas dos objetivos do trabalho, mas essas soluções falham em alguns pontos que se podem considerar importantes para o sistema a desenvolver. Nas tabelas 2.1 e 2.2, podemos observar a comparação entre as soluções analisadas, vendo a área de foco, os utilizadores alvo, os principais objetivos de cada uma das soluções analisadas e as suas diferentes valências.

A solução *Lifelines* é a solução que se encontra mais próxima do objetivo deste trabalho, pois esta solução mostra ao utilizador o historial clínico de um utente identificando com diferentes níveis a relevância dos eventos (com recurso a cores e com

diferentes espessuras de linha), permite também a visualização com maior detalhe de cada evento e mostra relações diretas entre eventos.

O sistema *CLEF* tem alguns aspetos importantes relativamente ao que se pretende fazer neste projeto, nomeadamente a identificação de relações causa-efeito, enquanto na parte da visualização da informação clínica, mais nomeadamente na representação da mesma, este sistema encontra-se mais afastado do objetivo do projeto.

KNAVE-II é um sistema que permite ao utilizador a observação, em linhas do tempo separadas, com mais detalhe cada um dos pontos existentes no processo clínico do utente, não permitindo que seja visualizado o processo de uma forma geral para que o profissional possa rapidamente saber o estado de saúde do utente.

No caso do sistema *TimeLine*, este permite ao utilizador visualizar a informação do utente *on-demand* conforme a árvore, que contém a informação organizada, é expandida. Com isto podemos afirmar que este sistema não permite que o utilizador visualize o estado de saúde do utente sem antes andar a navegar na árvore de informação.

O caso da solução *AsbruView* é um exemplo da aplicação de uma linha do tempo na área da saúde, mas esta solução tem como foco a criação e gestão de planos de tratamento, com isto esta solução deixa de estar apta para o problema proposto, pois como o foco é a criação de planos de tratamento deixa de ser possível ter informação relativa a outros aspetos do processo clínico, por exemplo resultados de exames e o histórico clínico.

A solução *SystemOne - Patient Timeline* é uma solução que foi implementada num sistema de processo clínicos eletrónicos (EHR), mas esta solução apenas permite ao utilizador visualizar o histórico clínico ao longo do tempo através da visualização de cada um dos itens, do processo clínico, em linhas do tempo separadas. Com isto não é possível ao utilizador visualizar de forma rápida o estado clínico do utente sendo necessário andar a alternar entre os diversos itens do processo.

Para finalizar, pretende-se realçar que a solução a desenvolver deverá colmatar as falhas das soluções aqui apresentadas, uma vez que no decorrer deste estudo a ideia que se pretende desenvolver foi enriquecida com pequenos aspetos que se encontram presentes nas soluções analisadas e que serão uma mais valia para o projeto. Também, será uma mais valia para o projeto, o facto de a nossa solução utilizar técnicas de reconhecimento de padrões que permitam ao utilizador, o profissional clínico, ter conhecimento sobre a possibilidade do utente em análise poder vir a contrair uma determinada patologia, ou não, sendo que os resultados obtidos podem ser interpretáveis pelo mesmo. Pretende-se, assim, oferecer uma boa ferramenta de trabalho para o utilizador, melhorando a sua capacidade de resposta aos vários problemas/desafios com que se depara.

Tabela 2.1: Comparação das diferentes soluções analisadas

Soluções	Área de aplicação	Utilizadores Alvo	Principais Objetivos
<i>Lifelines</i>	Clínica/Judicial/Militar	Profissionais clínicos; Profissionais judiciais; Militares	Visualização do histórico pessoal (Clínico, Judicial, Militar)
<i>CLEF</i>	Clínica	Profissionais clínicos; Investigadores Biomédicos	Tratamento e visualização dos processos clínicos com extração de informação
<i>KNAVE-II</i>			Visualização e exploração dos dados temporais em diferentes níveis de abstração temporal
<i>Timeline</i>		Profissionais clínicos	Processamento da informação presente em EHR's e visualização
<i>AsbruView</i>			Criação e estruturação de planos de tratamento
<i>SystemOne - Patient Timeline</i>			Visualização da informação presente no processo clínico do utente

Tabela 2.2: Comparação do funcionamento dos diferentes sistemas analisados

Soluções	Mostra a informação estruturada	Dados agregados numa só linha do tempo	Extração de conhecimento	Apresenta agendamentos futuros	Dados relativos ao presente e passado
<i>Lifelines</i>	✗	✓	✗	✗	✓
<i>CLEF</i>	✓	✓	✓	✗	✓
<i>KNAVE-II</i>	✓	✗	✓	✗	✓
<i>Timeline</i>	✓	✓	✓	✗	✓
<i>AsbruView</i>	✗	✓	✗	✓	✓
<i>SystemOne - Patient Timeline</i>	✓	✗	✗	✗	✓
<i>MedicineOne Visualização Temporal</i>	✓	✓	✓	✓	✓

Capítulo 3

Gestão de Projeto

Neste capítulo pretende-se explicar ao leitor como foi desenvolvido o projeto, identificando a metodologia de trabalho, o planeamento efetuado do projeto e os processos de engenharia de *software* utilizados.

3.1 Metodologia de trabalho

Os projetos de software devem seguir uma metodologia de desenvolvimento de forma a que exista uma boa gestão dos mesmos e controlo sobre o processo de desenvolvimento. Para o desenvolvimento de software existe um vasto leque de metodologias passíveis de serem utilizadas, embora existam algumas que se adaptem melhor a um determinado projeto do que outras metodologias.

Considerando que este projeto tem em conta a visão do utilizador, com uma equipa de desenvolvimento constituída apenas pelo estagiário e também tendo em conta a duração do estágio, optou-se pela escolha de uma metodologia de desenvolvimento iterativa e incremental de forma a permitir uma melhor articulação e obtenção de *feedback* do cliente.

Neste projeto está a ser utilizada a metodologia de trabalho *Feature-Driven Development* (FDD) [21–23], foi escolhida esta metodologia pois vai ao encontro do método de trabalho da equipa na qual o estagiário se encontra envolvido, embora o estagiário esteja a trabalhar sozinho no projeto. Este processo de desenvolvimento é um processo iterativo e incremental que permite que ao longo do seu desenvolvimento seja possível efetuar entregas faseadas do produto, tendo como objetivo principal o produto como um todo.

Durante o decorrer do desenvolvimento da linha do tempo foi adotada uma postura de *scrum*, uma vez que este desenvolvimento será feito com recurso a *sprints* com uma duração de duas semanas onde serão realizadas as respetivas *features* atribuídas a essa fase de desenvolvimento. No início/final de cada *sprint* é realizada uma reunião de forma a apresentar o progresso do projeto e as tarefas a realizar no *sprint* seguinte. Para a segunda parte deste projeto também foi mantida a postura de *scrum* mantendo os *sprints* com a duração de duas semanas. Sendo apresentado o progresso dessas semanas numa reunião a realizar no fim de cada *sprint*.

No primeiro semestre decorreram as primeiras fases desta metodologia de trabalho, em que consistiu no desenvolvimento do modelo global do projeto, o levantamento dos requisitos, a identificação da lista de *features* a desenvolver e o planeamento das fases seguintes.

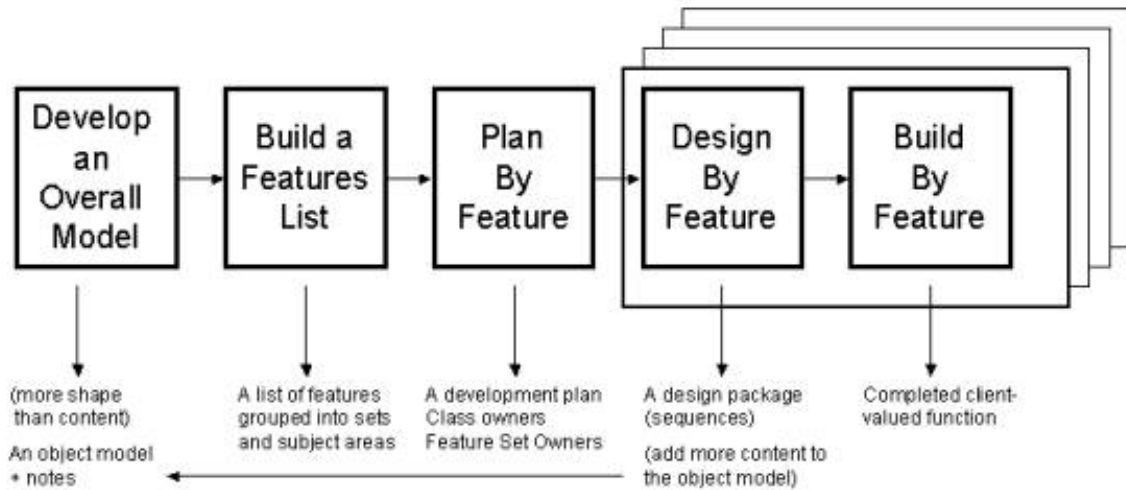


Figura 3.1: *Feature-Driven Development* (Adaptado a partir de [21]).

Após a identificação das *features*, durante a criação do plano de desenvolvimento estas foram priorizadas para que o seu desenvolvimento seja efetuado pelas mais importantes para o projeto, trazendo mais valor para o objetivo final.

3.2 Planeamento

Nesta secção é apresentado o planeamento realizado para o desenvolvimento do projeto ao longo dos dois semestres.

Durante o planeamento foi necessário a realização de estimativas de duração para as tarefas a realizar. Para esta estimação optou-se por utilizar a técnica *Tree-Point Estimation*, como especificado na secção 3.3 - Processos de engenharia.

3.2.1 Primeiro semestre

No decorrer do primeiro semestre do estágio foram realizadas as tarefas apresentadas na figura 3.2. Para o decorrer do primeiro semestre foram consideradas as tarefas do estado da arte e da avaliação intermédia pois estas tarefas não fazem diretamente parte do projeto em desenvolvido na empresa mas fazem parte do estágio, tendo sido o planeamento efetuado para o estágio e não só para o projeto desenvolvido. Este projeto no primeiro semestre teve como principal foco a análise, a documentação e a experimentação.

De seguida encontra-se o diagrama de *Gantt*, figura 3.3, referente ao planeamento inicial do primeiro semestre, sendo que a seguir na figura 3.4 encontra-se o diagrama de *Gantt* final do primeiro semestre, que nos apresenta as tarefas desenvolvidas no decorrer deste.

	Nome	Duração	Ínicio	Fim
1	<input type="checkbox"/> Estágio Primeiro Semestre	89d	30/09/2015	01/02/2016
2	Início Estágio	1d	30/09/2015	30/09/2015
3	Estado da arte	29d	01/10/2015	10/11/2015
4	Experimentação e Def. Tecnologias	30d	11/11/2015	22/12/2015
5	Especificação de Requisitos	20d	02/11/2015	27/11/2015
6	Desenvolvimento da Arquitetura	20d	07/12/2015	01/01/2016
7	Especificação da Arquitetura	6d	01/01/2016	08/01/2016
8	Início do desenvolvimento	11d	15/01/2016	29/01/2016
9	<input type="checkbox"/> Avaliação Intermédia	79d	14/10/2015	01/02/2016
10	Escrita Relatório	72d	14/10/2015	21/01/2016
11	Entrega Relatório Intremédio	1d	22/01/2016	22/01/2016
12	Preparação Defesa	5d	25/01/2016	29/01/2016
13	Defesa Intermédia	1d	01/02/2016	01/02/2016

Figura 3.2: Lista de tarefas planeadas para o primeiro semestre

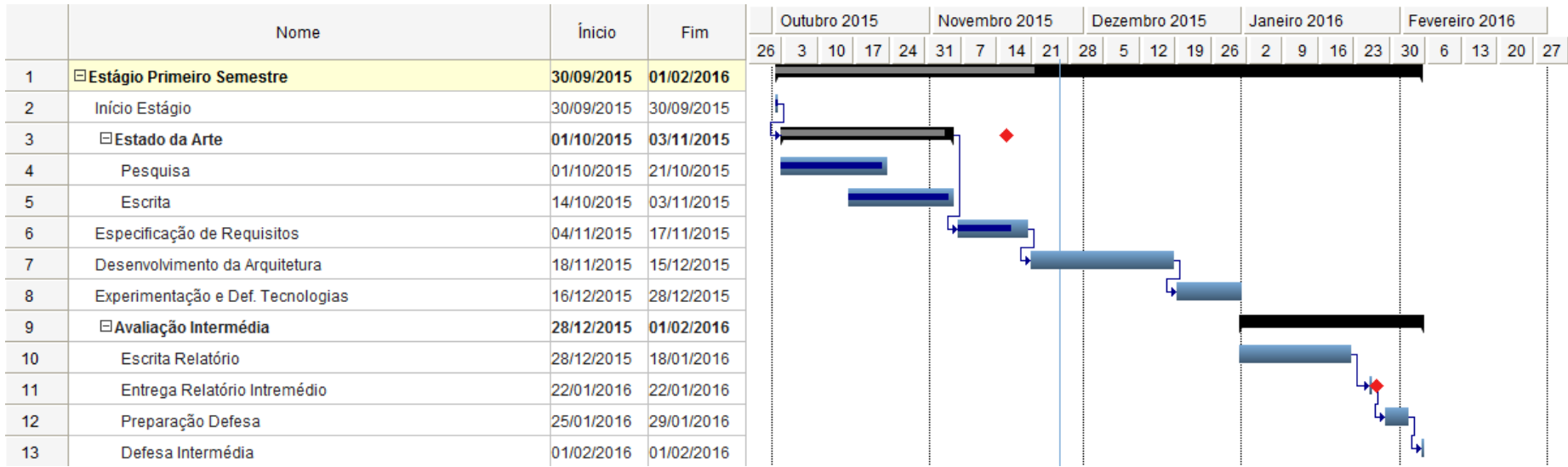


Figura 3.3: Diagrama de *Gantt* inicial relativo ao primeiro semestre

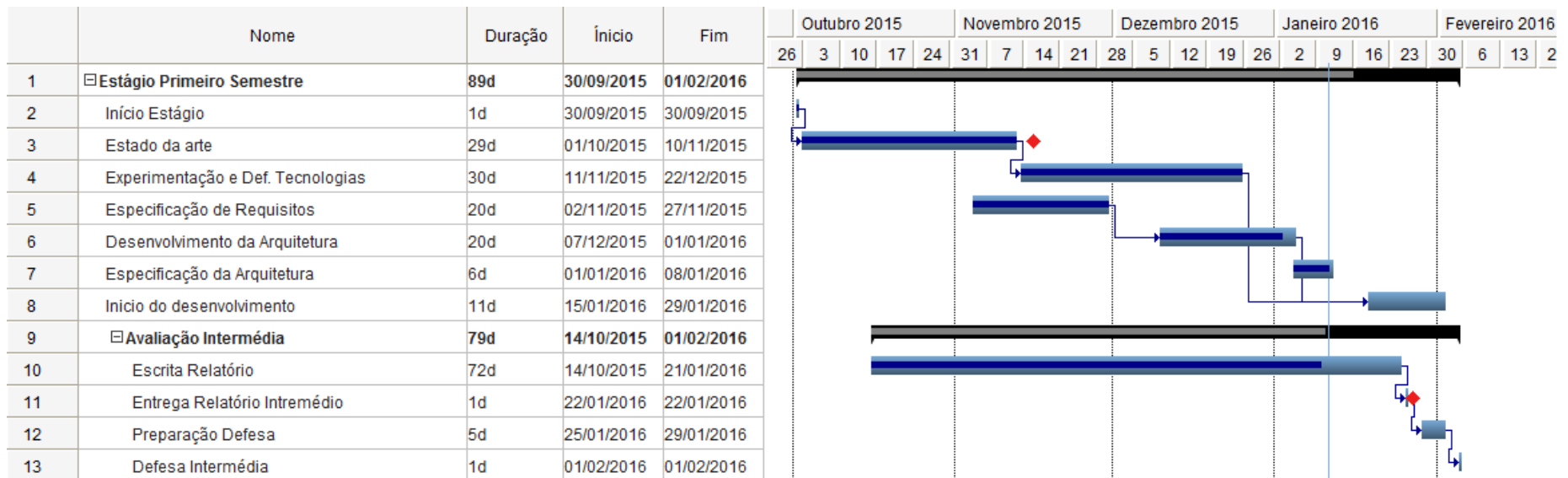


Figura 3.4: Diagrama de *Gantt* final relativo ao primeiro semestre

Como é possível observar pela comparação dos diagramas de *Gantt*, apresentados anteriormente, não ocorreram grandes derrapagens no planeamento do primeiro semestre. Os pontos que sofreram uma alteração maior foram a escrita do relatório de estágio e a fase de experimentação e definição de tecnologias. A escrita de relatório foi iniciada numa fase mais inicial do estágio para que a fase final, deste semestre, não fosse sobrecarregada com esta tarefa, o que poderia levar a atrasos das outras tarefas a realizar. Quanto à experimentação e definição das tecnologias, também se optou por realizar numa fase mais inicial do projeto de forma a que fosse possível analisar a complexidade do mesmo e ter uma melhor noção do que seria possível de desenvolver com o componente selecionado.

3.2.2 Segundo semestre

Para o segundo semestre foram planeadas as tarefas apresentadas na figura 3.5, sendo que está previsto o desenvolvimento da linha do tempo e o estudo e implementação de um mecanismo que permita efetuar a análise do processo clínico e decidir se existe a possibilidade do paciente poder contrair diabetes, mostrando depois esse resultado ao utilizador na linha do tempo com a representação das relações entre os eventos que levaram o sistema a decidir.

	Nome	Duração	Íncio	Fim
1	☐ Estágio 2º Semestre	113d	25/01/2016	29/06/2016
2	☐ Desenvolvimento Protótipo da Linha do tempo	60d	25/01/2016	15/04/2016
3	Fase de desenvolvimento 1	10d	25/01/2016	05/02/2016
4	Fase de desenvolvimento 2	10d	08/02/2016	19/02/2016
5	Fase de desenvolvimento 3	10d	22/02/2016	04/03/2016
6	Fase de desenvolvimento 4	10d	07/03/2016	18/03/2016
7	Fase de desenvolvimento 5	10d	21/03/2016	01/04/2016
8	Fase de desenvolvimento 6	10d	04/04/2016	15/04/2016
9	☐ Estágio Fase 2 - Relacionamento entre eventos	54d	04/04/2016	16/06/2016
10	Feature Extraction Fase 1	10d	04/04/2016	15/04/2016
11	Feature Extraction Fase 2	10d	18/04/2016	29/04/2016
12	Feature Selection	10d	02/05/2016	13/05/2016
13	Testes Classificadores (Pré-Processamento)	10d	12/05/2016	25/05/2016
14	Testes Classificadores (Modelo Final)	10d	26/05/2016	08/06/2016
15	Integração com o protótipo desenvolvido	10d	03/06/2016	16/06/2016
16	Relatório Final	76d	16/03/2016	29/06/2016

Figura 3.5: Lista de tarefas planeadas para o segundo semestre

Para o desenvolvimento da linha do tempo decidiu efetuar-se a divisão em 6 fases de desenvolvimento por onde foram distribuídos os requisitos a desenvolver. Na lista seguinte podemos verificar os requisitos que foram atribuídos a cada uma das fases, para efetuar essa distribuição teve-se em conta a tabela da priorização e das estimativas dos mesmos que se encontram nos Apêndices C e D.

- **Fase de desenvolvimento 1** - Para esta primeira fase foram selecionados os requisitos de ordem #01 a #05, sendo estes o *Ge_US05*, *Ge_US09*, *Ge_US01*, *Ge_US07* e *Ge_US08*.
- **Fase de desenvolvimento 2** - Nesta fase ficaram os requisitos de ordem #06 a #08 sendo que estes são os requisitos: *Pa_US01*, *Ge_US03* e *Pa_US02*.
- **Fase de desenvolvimento 3** - A fase 3 ficou com os requisitos *Pa_US06*, *Pr_US03*, *Pr_US06*, *Pa_US05*, *Pa_US03* e *Fu_US04*, sendo estes os requisitos de ordem #09 a #14.
- **Fase de desenvolvimento 4** - Os requisitos de ordem #15 a #18 foram atribuídos a esta fase de desenvolvimento e estes são os requisitos *Fu_US01*, *Fu_US03*, *Fu_US05* e *Pr_US05*.
- **Fase de desenvolvimento 5** - Nesta quinta fase ficaram os requisitos de ordem #19 a #21 (*Ge_US06*, *Pr_US01* e *Pr_US07*).
- **Fase de desenvolvimento 6** - Esta última fase ficou com os restantes requisitos, sendo estes os de ordem #22 a #24 (*Ge_US04*, *Fu_US02* e *Pr_US02*).

Têm-se a destacar que a fase de desenvolvimento 6, devido às alterações ao plano de estágio e a opção de realizar o estudo e implementação de análise e deteção da possibilidade de um utente contrair diabetes e os fatores que levam a essa decisão, levou a que esta fase de desenvolvimento fosse colocada de lado para trabalho futuro de forma a que a segunda fase do estágio pudesse avançar. Esta segunda fase vai de encontro com o requisito *Pa_US04* uma vez que possibilita mostrar as relações entre eventos, neste caso especificamente entre o resultado da decisão e os fatores que levaram o sistema a tomar a mesma.

Na figura 3.6 encontra-se o respetivo diagrama de *Gantt* referente ao segundo semestre deste estágio.

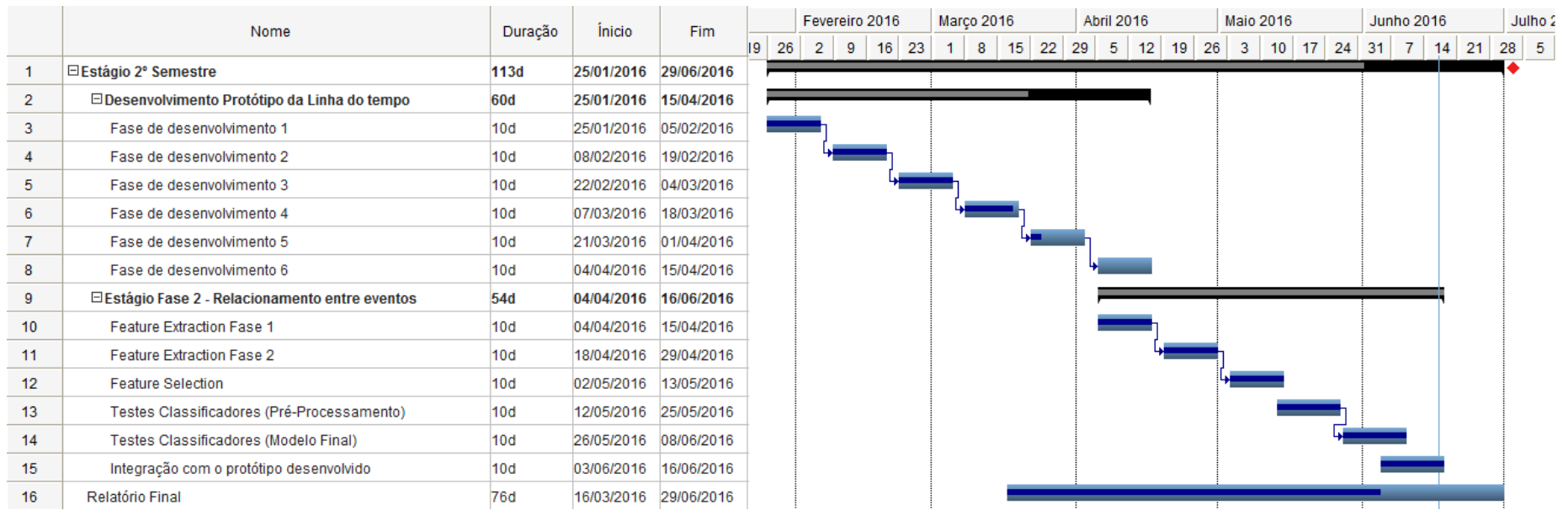


Figura 3.6: Diagrama de *Gantt* referente ao segundo semestre

3.3 Processos de engenharia

Durante o decorrer do desenvolvimento do projeto foram utilizado alguns processos de engenharia de *software* [24], que permitiram uma melhor gestão e organização do trabalho desenvolvido e a desenvolver, de seguida são apresentados os diferentes processos utilizados no projeto.

- **Gestão de Código** - Durante o desenvolvimento foi utilizado o sistema de gestão de código *Team Foundation Server*(TFS), que veio permitir uma melhor gestão do desenvolvimento do projeto, assim como o controlo das versões.
- **OnTime**: O sistema *OnTime* é a aplicação utilizada, pela empresa, para a gestão das *features*, *defects*, *incidents* e das tarefas a realizar. As diferentes tarefas são distribuídas a cada um dos elementos da equipa pelo chefe de equipa. Este sistema foi utilizado na fase de desenvolvimento do projeto tendo sido adicionadas, no sistema, as diferentes *features* do trabalho a desenvolver e as mesmas atribuídas ao estagiário.
- **Tree-Point Estimation** - Foi escolhida esta técnica tendo em conta as diferentes tarefas a realizar no decorrer do projeto em conjunto com a metodologia de trabalho selecionada, sendo esta uma metodologia baseada nas *features* do projeto. Para efetuar a estimacão utilizou-se a seguinte fórmula:

$$Expected = (1 \times Best\ Case + 4 \times Most\ Likely + 1 \times Worst\ Case)/6$$

A tabela com as estimativas realizadas para cada um dos requisitos pode ser consultada no Apêndice D.

- **Reuniões de equipa**: São realizadas reuniões internas de equipa semanalmente, com o propósito de acompanhamento do trabalho em desenvolvimento por cada elemento da equipa. Nestas reuniões cada elemento da equipa apresenta oralmente o trabalho que foi realizado na semana anterior e o que tem planeado para os dias seguintes. Também são discutidas formas de como resolver os diferentes problemas com que cada um se depara.
- **Reuniões com o cliente**: Neste projeto têm-se como cliente o fundador da empresa e criador da ideia, sendo que se realizam reuniões semanais com o mesmo para o acompanhamento do desenvolvimento do projeto, esclarecimento de dúvidas e discussão de ideias.
- **Sharepoint**: Este sistema é o sistema utilizado na empresa para o armazenamento de documentos referentes aos projetos. Durante o decorrer do estágio foi utilizado para a publicação dos diferentes documentos internos que foram sendo produzidos pelo estagiário.

3.4 Gestão de riscos

Como todos os projetos de *software*, também num projeto de estágio existem riscos que podem colocar em causa o sucesso do mesmo. Neste ponto são apresentados os riscos e como foi efetuada a sua gestão [24].

Tabela 3.1: Definição dos níveis de probabilidade de ocorrência.

Denominação	Probabilidade
Raro	<20%
Pouco Provável	20% - 50%
Provável	50% - 80%
Muito Provável	>80%

Na tabela anterior são apresentados os diferentes níveis de probabilidade definidos e quais os valores associados a cada um. De seguida, encontra-se a definição dos diferentes níveis de impacto associados a cada um dos riscos que indicam o que impacto de um risco caso este venha a se verificar.

- **Baixo** - A conclusão do projeto não se encontra em risco.
- **Médio** - É possível concluir o projeto dentro do esforço previsto.
- **Elevado** - Para se conseguir concluir o projeto é necessária a aplicação de esforço extra.
- **Crítico** - Compromete seriamente a conclusão do projeto.

Para além da probabilidade de ocorrência de um determinado risco e do impacto do mesmo no projeto também é definida uma janela de tempo que nos pode ajudar a perceber quando é que um determinado risco poderá ocorrer no nosso projeto. De seguida são apresentados os termos definidos para a janela de tempo e o seu significado:

- **Curto-Prazo:** Os riscos classificados com este termo podem ocorrer na fase inicial do projeto, até às primeiras semanas de desenvolvimento.
- **Médio-Prazo:** Este termo define os riscos que podem afetar o projeto na fase principal do desenvolvimento, ou seja, no decorrer do desenvolvimento do componente.
- **Longo-Prazo:** Os riscos que forem classificados com a janela de tempo a longo prazo, podem afetar o projeto na fase final do mesmo, a fase de integração, ou posteriormente a esta fase.

3.4.1 Riscos

De seguida são apresentados e descritos os riscos identificados que podem vir a afetar o projeto.

Curva de aprendizagem elevada

ID: R01

Descrição: No decorrer do projeto foram utilizadas algumas tecnologias com as quais o estagiário ainda não tinha experiência de desenvolvimento, o que pode levar a que a curva de aprendizagem seja maior que o esperado. Este facto pode levar a que o projeto sofra atrasos devido à falta de experiência com as tecnologias.

Plano de mitigação: Início da fase de experimentação das tecnologias mais cedo para que o estagiário se possa ambientar às mesmas.

Janela de tempo: Médio-Prazo

Impacto: Médio

Probabilidade: Pouco Provável

Subestimação do tempo das tarefas

ID: R02

Descrição: Sendo que o desenvolvimento das tarefas efetuado por apenas uma pessoa, um atraso na realização de uma determinada tarefa, resultante de uma subestimação da mesma, pode ter um impacto significativo na conclusão do projeto.

Plano de mitigação: Como plano de mitigação optou-se por uma priorização das tarefas para que as tarefas essenciais sejam as primeiras a serem realizadas.

Janela de tempo: Médio-Prazo

Impacto: Elevado

Probabilidade: Provável

Aquisição das licenças do *software* da *DevExpress*, não estar concluída aquando do início do desenvolvimento

ID: R03

Descrição: Para o desenvolvimento deste projeto foi necessário a utilização dos componentes de *User Interface*(UI), da *DevExpress*, da última versão. O processo de aquisição das novas licenças da *DevExpress* é um processo que pode levar a algum tempo a concretizar e pode sofrer atrasos. Isto pode levar a que o desenvolvimento possa parar ou atrasar.

Plano de mitigação: Utilização da versão *trial* (30 dias) dos componentes e em caso da licença expirar requerer à *DevExpress* uma extensão da licença *trial*, até à conclusão da aquisição das licenças.

Janela de tempo: Curto-Prazo

Impacto: Crítico

Probabilidade: Muito Provável

Inadaptação da arquitetura do componente à arquitetura do *MedicineOne 8*

ID: R04

Descrição: Como o componente a desenvolver deverá ser integrado no *MedicineOne 8*, a arquitetura deste deverá ser similar à arquitetura do sistema onde será integrado. Com isto podem surgir problemas na compatibilidade das duas arquiteturas, podendo por em causa as funcionalidades do módulo.

Plano de mitigação: Estudo prévio da arquitetura do sistema onde será integrado o componente.

Janela de tempo: Médio-Prazo

Impacto: Crítico

Probabilidade: Raro

Mau desempenho da metodologia de classificação

ID: R05

Descrição: Como a segunda componente deste estágio utiliza uma metodologia de classificação, esta pode ter uma performance abaixo do esperado o que leva a que a mesma não possa ser utilizada por ter uma má performance.

Plano de mitigação: Recolher mais características a partir das base de dados originais e aplicar mais técnicas de pré-processamento de dados de forma a melhorar a performance do classificador.

Janela de tempo: Médio-Prazo

Impacto: Elevado

Probabilidade: Pouco Provável

Tabela 3.2: Tabela de exposição dos riscos

		Impacto			
		Baixo	Médio	Elevado	Crítico
Probabilidade	Muito Provável			R03	
	Provável			R02	
	Pouco Provável		R01	R05	
	Raro				R04

3.4.2 Mitigação de riscos

A tabela 3.2 representa a matriz de exposição dos riscos identificados, tendo em conta a probabilidade de ocorrência e o impacto de cada um dos riscos.

Como estratégia de mitigação dos riscos optou-se pela priorização dos riscos com recurso à abordagem *Pareto Top-N* [25] em que se pretende mitigar 33% dos riscos identificados tendo em conta os parâmetros do impacto e da probabilidade, em que o impacto seja igual ou superior à classificação de "elevado" e a probabilidade seja superior a 50% (provável ou muito provável). Com isto podemos dizer que os principais riscos a serem mitigados são os riscos **R02** e **R03**, pois se encontram dentro da zona definida pelos parâmetros anteriormente descritos.

Riscos verificados

Já no decorrer do primeiro semestre com o início da fase de experimentação existiu um risco que se veio a verificar. Esse risco foi o risco com o *id* **R03**, em que o processo de aquisição das licenças de *software* da nova versão *DevExpress* sofreu alguns atrasos. Dada esta ocorrência foi necessário executar o plano de mitigação que consistiu a utilização de uma versão *trial* da *DevExpress* e efetuar o pedido de extensão do *trial* para a utilização dos respetivos componentes.

Para além do risco **R03** também se verificou o risco **R02** numa fase mais avançada do projeto no decorrer do desenvolvimento da linha do tempo, onde se verificou que seria necessário parar o desenvolvimento no decorrer da fase 5, não realizando a fase 6, uma vez que os principais requisitos já se encontravam terminados, e avançar para a segunda parte do estágio de forma a que no fim do mesmo já fosse possível apresentar alguns resultados dessa parte.

CAPÍTULO 3. GESTÃO DE PROJETO

Capítulo 4

Análise de Requisitos

Neste capítulo será apresentada a análise de requisitos que foi elaborada, que consistiu na identificação dos atores do sistema, as *User Stories*(US), os *mockups* das interfaces que foram realizados, a enumeração dos requisitos funcionais do sistema, os atributos de qualidade e por fim as diferentes restrições a que o projeto se encontra sujeito.

4.1 Atores do sistema

Neste ponto será descrito o papel dos atores que irão interagir com o módulo, sendo estes os mesmos atores que têm acesso na aplicação *MedicineOne 8* ao processo clínico.

Apesar de este sistema poder ser utilizado pelos vários profissionais de saúde, médicos, enfermeiros, entre outros, optou-se por generalizar o ator pois ambos os profissionais terão o mesmo tipo de interação com o módulo. Sendo que as limitações de acesso a zonas do *software* são personalizáveis dentro das diferentes unidades de saúde. Podendo, por exemplo, em algumas unidades de saúde um enfermeiro chefe ter acesso a operações que em outras unidades de saúde apenas os médicos tenham. Podemos assim nomear o nosso ator de **profissional de saúde**.

4.2 *User Stories*

A especificação das diferentes funcionalidades do projeto foram descritas recorrendo às *User Stories* (US), isto porque as US permitem fazer uma descrição breve das funcionalidades do ponto de vista do utilizador/cliente. A utilização das US teve em vista o facto da metodologia do projeto ser uma metodologia ágil e destas serem facilmente estimáveis, o que vem permitir um melhor planeamento do projeto. Além destes aspetos as US são simples de construir e facilmente podem ser mapeadas em testes de aceitação para validar as funcionalidades implementadas. Para a especificação das US foi utilizada a seguinte estrutura [26–29]:

- **Enquanto** <tipo_utilizador>
- **Quero** <ação_realizar>
- **Para** <benefício>

Um outro aspeto importante, para além da definição das US, é a priorização das mesmas, que vai permitir à equipa de desenvolvimento saber quais as US que serão desenvolvidas em primeiro lugar. De seguida encontram-se identificados os termos utilizados para a priorização e o nível de importância de cada termo utilizado.

- **Must:** Os requisitos com esta classificação têm de ser implementados obrigatoriamente, pois sem estes requisitos o bom funcionamento do sistema encontra-se comprometido.
- **Nice:** Os requisitos identificados com esta classificação devem ser implementados, sendo que caso não sejam implementados não comprometem o funcionamento do sistema.
- **Wishful:** Os requisitos que tenham a classificação *Wishful* são requisitos com um carácter opcional uma vez que não se espera que sejam implementados.

4.3 Requisitos funcionais

Os requisitos funcionais levantados foram divididos em quatro categorias. Esta divisão teve por base o relacionamento de cada requisito funcional com o componente a desenvolver. Caso o requisito seja relativo apenas à secção do *Presente*, este é inserido na categoria do presente. Caso o requisito esteja relacionado com a secção do *Passado*, este será inserido na respetiva categoria. E à semelhança dos requisitos das duas secções descritas anteriormente, os requisitos relativos à secção do *Futuro* também serão atribuídos à categoria do futuro. Todos os outros requisitos que sejam comuns às várias secções estarão inseridos na categoria dos requisitos gerais ao sistema.

Para a especificação dos requisitos funcionais foram utilizadas *User Stories*, com a estrutura definida no ponto anterior. De seguida encontra-se a listagem sumária dos requisitos levantados em cada uma das categorias identificadas. No Apêndice B encontram-se as tabelas com a especificação dos requisitos funcionais aqui apresentados, com as US e a priorização dos mesmos. No Apêndice C pode ser consultada uma tabela onde os requisitos se encontram ordenados pela ordem a implementar.

4.3.1 Requisitos gerais ao sistema

Os requisitos gerais ao sistema tem os seus id's começados por "Ge_US0x".

- **Ge_US01** - Apresentar a secção do Presente na abertura do módulo
- **Ge_US02** - Visualizar as notificações dos elementos do agregado familiar
- **Ge_US03** - Mostrar sumário de um evento ao passar com o cursor do rato (*tooltip*)
- **Ge_US04** - Ao fazer duplo-clique sobre um evento mostrar, numa janela, a informação detalhada
- **Ge_US05** - Alternar entre os diferentes momentos
- **Ge_US06** - Introduzir marcadores para identificar ocorrências relevantes

- **Ge_US07** - Modificar o código de cores
- **Ge_US08** - Selecionar grupos de eventos visíveis
- **Ge_US09** - Alternar entre os elementos familiares e apresentar a respetiva informação

4.3.2 Requisitos do Presente

Os id's dos requisitos relativos ao momento do presente começam por "Pr_US0x".

- **Pr_US01** - Apresentar, inicialmente, a informação mais atual de cada setor do processo clínico do utente dos últimos 3 meses
- **Pr_US02** - Apresentar os valores em faixas separadas segundo a sua classificação (Acima do normal, normal e abaixo do normal)
- **Pr_US03** - As 3 faixas onde são representados os valores devem ter as seguintes cores, faixa central a verde, a faixa superior e a inferior devem ter um fundo branco
- **Pr_US04** - Mostrar alertas do utente e permitir o seu processamento
- **Pr_US05** - Apresentar dados com recurso a código de cores, pelo tipo de informação
- **Pr_US06** - Personalizar o intervalo de tempo dos dados a visualizar
- **Pr_US07** - Listar eventos que se encontrem ativos no momento

4.3.3 Requisitos do Passado

Nos requisitos relativos ao passado os id's começam por "Pa_US0x".

- **Pa_US01** - Apresentar o histórico clínico do utente
- **Pa_US02** - Apresentar dados com recurso a código de cores, pelo tipo de informação
- **Pa_US03** - Mostrar legenda do código de cores utilizado
- **Pa_US04** - Assinalar graficamente relacionamentos do tipo causa-efeito entre eventos
- **Pa_US05** - Deverá estar visível um *overview* de todos os eventos da linha do tempo
- **Pa_US06** - Permitir realizar operações de zoom-in e zoom-out

4.3.4 Requisitos do Futuro

Quanto aos requisitos do futuro os id's começam por "Fu_US0x".

- **Fu_US01** - Apresentar atividades agendadas para o futuro por ordem cronológica
- **Fu_US02** - Processar atividades futuras
- **Fu_US03** - Identificar as atividades com informação relativa ao seu estado (Agendada ou Planeada) e data
- **Fu_US04** - Permitir realizar operações de zoom-in e zoom-out
- **Fu_US05** - Apresentar as atividades que se encontram planeadas para o futuro

4.4 Prototipagem

Na fase inicial do projeto, conforme os requisitos foram sendo definidos os *mockups* iniciais foram sofrendo algumas alterações, sendo que estas ficaram a cargo do estagiário. De seguida serão apresentados os *mockups* finais do projeto, para as diferentes interfaces e para a especificação de alguns componentes presentes nas mesmas.

4.4.1 *Mockups* iniciais

Os *mockups* iniciais, que foram fornecidos pela equipa de design da empresa, encontram-se no Apêndice A.

4.4.2 *Mockups* finais

Neste ponto serão apresentadas as versões finais dos *mockups*, que surgiram das alterações discutidas ao longo das reuniões com o cliente e que eram realizadas após essas reuniões.

Interfaces principais

Os *mockups* aqui apresentados representam os referentes às três principais visualizações, o Presente, o Passado e o Futuro.

Presente

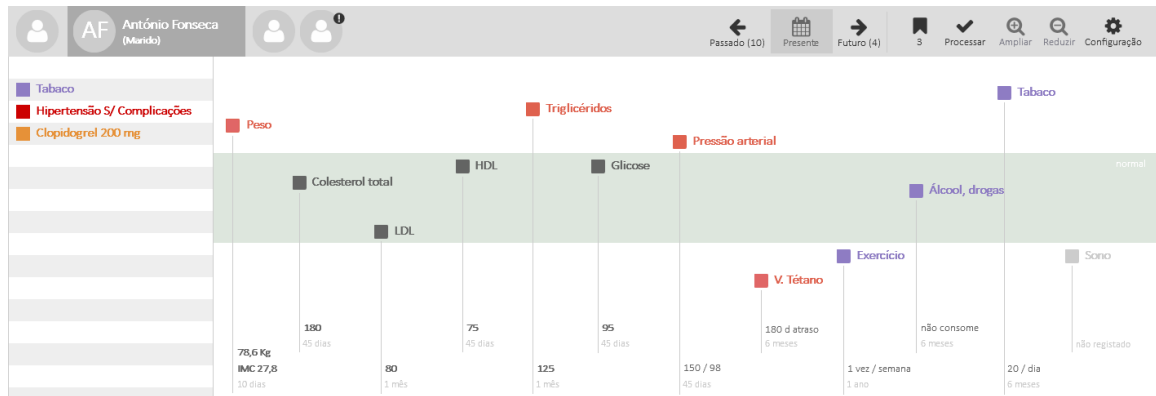


Figura 4.1: Mockup final para o presente

Passado

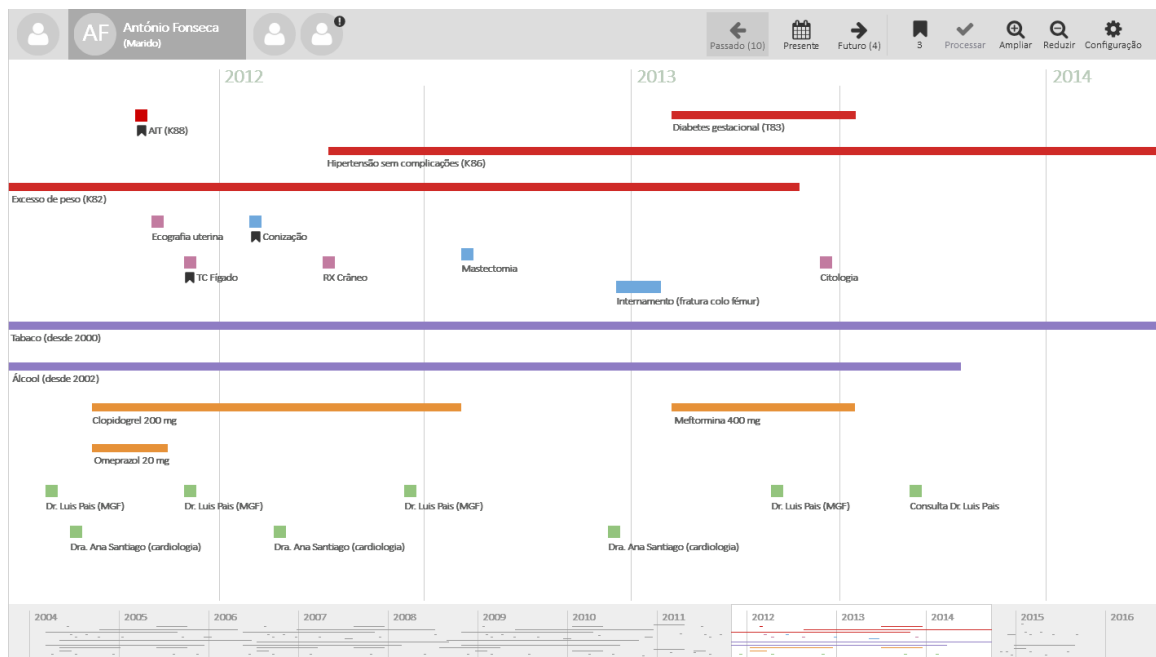


Figura 4.2: Mockup final para o passado

Futuro



Figura 4.3: *Mockup* final para o futuro

Outros componentes do módulo

Neste ponto são apresentados os *mockups* relativos aos diferentes componentes que complementam as diferentes interações do utilizador com o módulo.

Processar atividades futuras

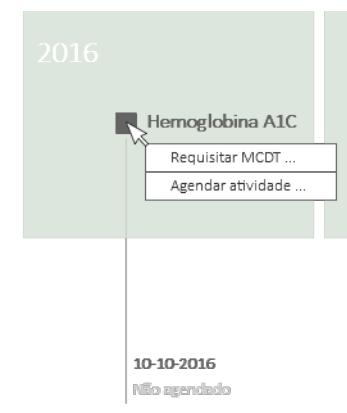


Figura 4.4: *Mockup* relativo à opção de processar atividade, existente na vista do futuro

Barra de ferramentas



Figura 4.5: Barra de ferramentas, que se vai encontrar presente em todas as vistas

Overview da linha do tempo



Figura 4.6: Overview da linha do tempo

Marcadores

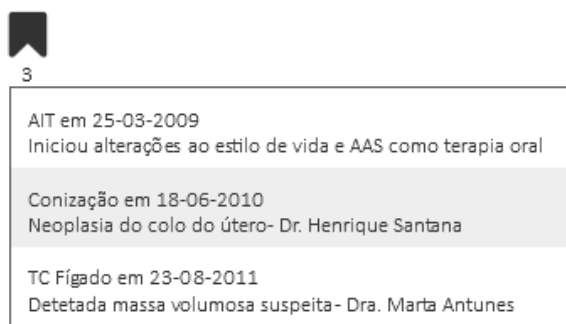


Figura 4.7: Lista com os itens marcados

Tooltip

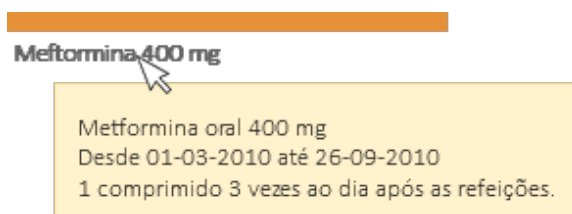


Figura 4.8: Exemplo de um tooltip com a informação a mostrar

Janela de configuração

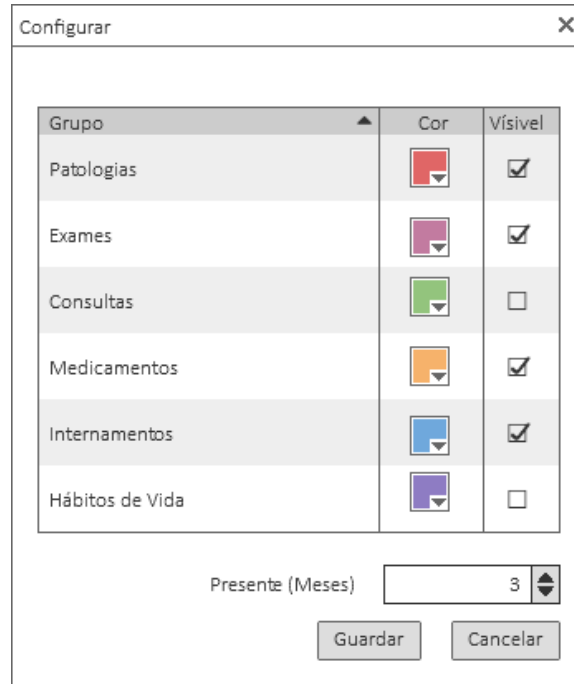


Figura 4.9: Janela de configuração

4.5 Restrições

Sendo que este projeto foi desenvolvido no âmbito de uma empresa, existem algumas restrições inerentes ao desenvolvimento do mesmo, que em caso de não serem identificadas corretamente podem invalidar o futuro do projeto. De seguida encontram-se listadas as restrições do projeto que foram identificadas.

4.5.1 Restrições técnicas

- Sendo que este projeto passa pela realização de um componente para integrar no produto da empresa o *MedicineOne 8*, têm-se como restrição técnica a utilização das tecnologias com que o produto está a utilizar, nomeadamente a utilização da linguagem de programação *C#* com a *.NET Framework 4.0* para o desenvolvimento.
- Para a realização do *design* da aplicação foi utilizado o pacote de componentes de UI da *DevExpress*¹ na medida em que todo o sistema utiliza estes componentes no seu design, no entanto para o componente da linha do tempo pode ser utilizado outro pacote de controlos de UI.
- Para além das tecnologias de desenvolvimento, como o projeto tem uma forte componente de UI, é necessário seguir as *guidelines* que a empresa tem para o desenvolvimento de interfaces gráficas, para que o componente desenvolvido se

¹<https://www.devexpress.com/>

incorpore no produto naturalmente, sem que este seja um componente forasteiro.

4.5.2 Restrições de negócio

- Seguir as normas de desenvolvimento de *software*, definidas pela empresa, para a garantia da qualidade do software desenvolvido.

4.5.3 Restrições legais

- Pela lei 12/2005 de 26 de Janeiro [30], o sistema deve garantir a sua conformidade conforme referido na lei supra citada. Para que os dados de saúde sejam confidenciais e o seu acesso seja restringido apenas a quem por lei lhe é permitido o devido acesso.

CAPÍTULO 4. ANÁLISE DE REQUISITOS

Capítulo 5

Arquitetura

Neste capítulo é apresentada inicialmente a arquitetura do sistema onde será integrado o componente a desenvolver. Posteriormente será apresentada a arquitetura do componente a desenvolver, assim como os modelos de dados necessários para o projeto. Por fim são apresentadas as tecnologias que serão utilizadas para o desenvolvimento do projeto.

Como este projeto será realizado com o intuito de integrar no *Medicione 8*, a sua arquitetura terá como base a atual do sistema onde será integrado, sendo esta aqui apresentada.

5.1 Arquitetura do sistema

De seguida é apresentada a arquitetura do sistema, sendo que é apresentada inicialmente a vista *Birds Eye* para que o leitor possa facilmente visualizar a composição do sistema. Posteriormente é apresentada uma vista de camadas e para finalizar é apresentada a perspectiva lógica. Após a apresentação do sistema onde o componente estará integrado é apresentada a arquitetura do componente que segue o *design pattern Model View Presenter*.

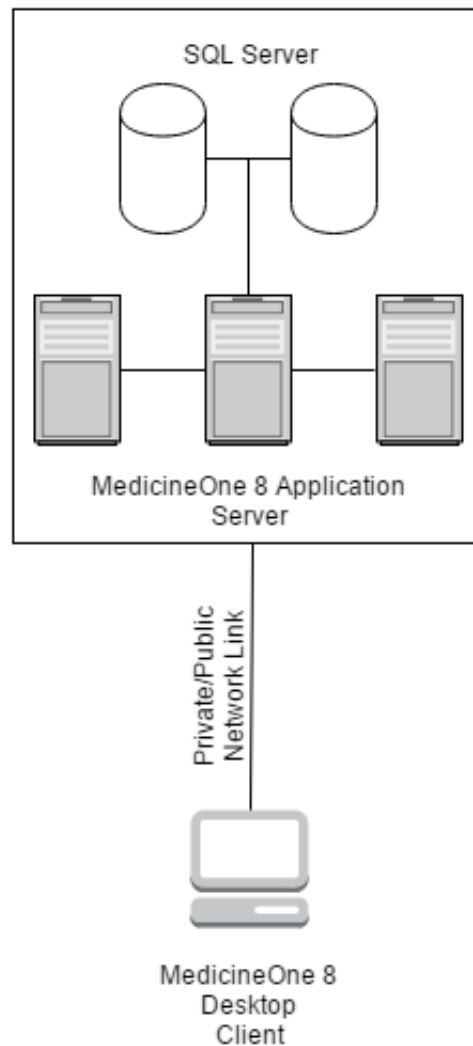
5.1.1 *Birds Eye View*

No diagrama apresentado na figura 5.1, é possível visualizar de uma forma geral a composição do sistema. A aplicação cliente pode ligar-se ao servidor através de uma ligação pública ou privada conforme a configuração física do sistema, se o servidor está ou não na rede do cliente.

Do lado do servidor temos uma divisão entre servidor de aplicação e as base de dados. O acesso por parte do cliente aos dados armazenados na base de dados é efetuado através do servidor de aplicação que fornece essa ligação.

O servidor de aplicação para além de fazer a ligação entre a base de dados e os clientes, também contém a lógica de negócio do sistema.

A base de dados (*SQL Server*), vai conter todos os dados que circulam no sistema, incluindo a informação dos utentes e dos utilizadores. Os servidores de base de dados apenas comunicam com o cliente indiretamente através dos servidores de aplicação onde se encontra a lógica de negócio.

Figura 5.1: *Birds Eye View*

5.1.2 Camadas e componentes do sistema

O sistema está dividido em 3 camadas. Na figura 5.2 podemos observar cada uma delas e os seus componentes. De seguida encontram-se especificadas cada uma das camadas com os respetivos componentes.

- ***MedicineOne Database (Data)*** - Nesta camada é onde se encontram localizadas as base de dados do sistema.
- ***MedicineOne Server (Business)*** - Nesta camada estará localizada a lógica da aplicação assim como toda a comunicação entre os clientes e as base de dados. A lógica da aplicação encontra-se aqui disponível através de *webServices* (WCF, ASP.NET).
- ***MedicineOne Client (Presentation)*** - Por fim temos a camada cliente que se encontra do lado do cliente da aplicação, cliente *desktop*. Nesta camada temos o nosso cliente da aplicação assim como uma base de dados que serve de *cache* melhorando assim performance da aplicação, pois assim os dados que menos alterações sofrem são armazenados do lado do cliente facilitando o seu

acesso e não tendo este a necessidade de estar a realizar pedidos ao servidor sempre que precisa de aceder a essa informação. A informação contida nesta *cache* local é não confidencial e não crítica. As tabelas locais são geridas com recurso ao *SQL Server Compact Edition*(SQLCe).

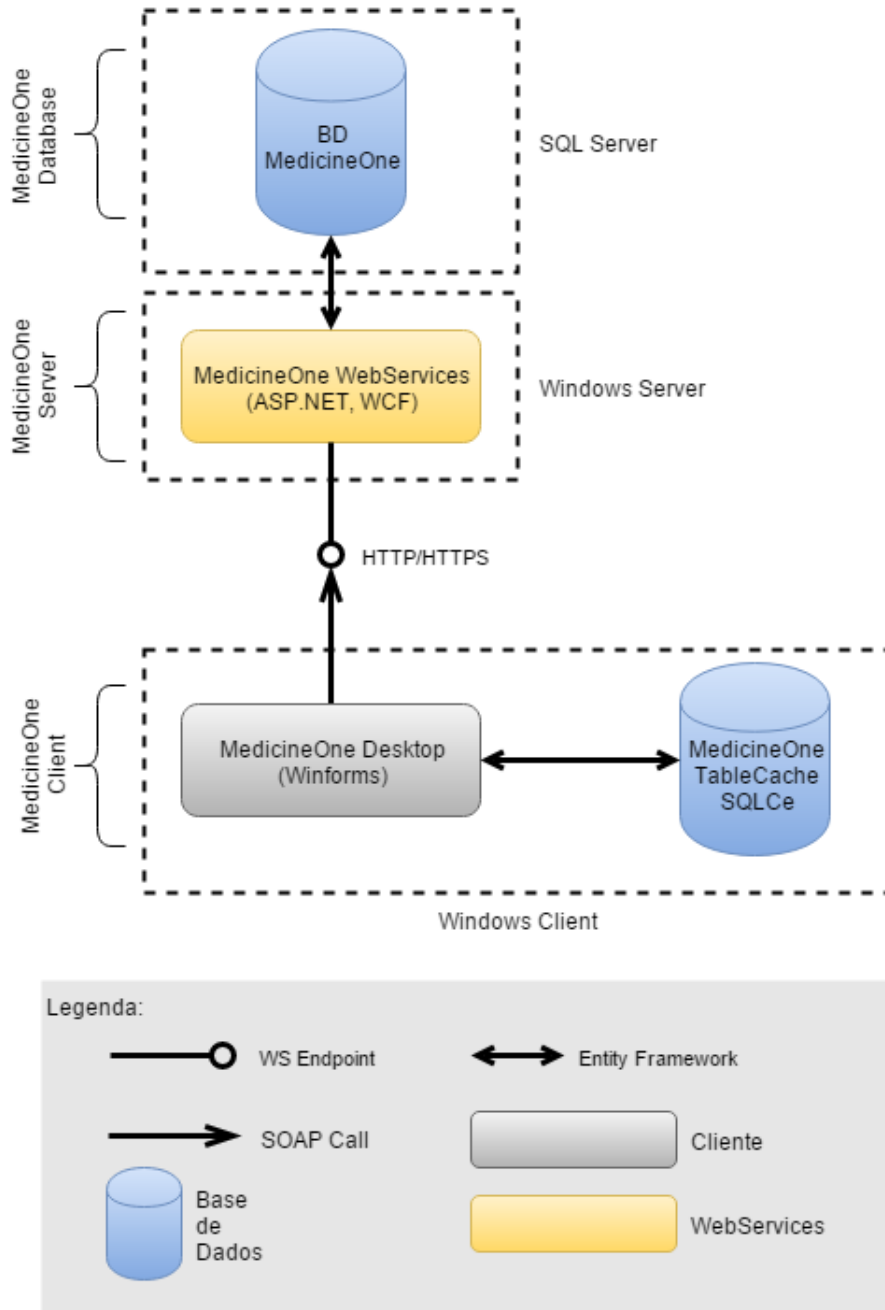


Figura 5.2: Diagrama de camadas e componentes

Também no diagrama podemos observar que os pedidos entre o cliente e os *web-services* são realizados com recurso à tecnologia *SOAP*, por via de HTTP/HTTPS.

A comunicação entre os módulos e as base de dados é efetuada com recurso a *Entity Framework*, efetuando assim o mapeamento dos dados nas bases de dados.

No diagrama 5.3 podemos observar os vários componentes do sistema que fazem parte do âmbito do estágio. Relativamente à linha do tempo, temos no servidor o *webservice ClinicalProcessTemporalViewService* que é o serviço que tem como principal objetivo a recolha e transmissão dos dados para a linha do tempo, *ClinicalProcessTemporalViewUserControl*. É também através deste serviço que são realizados os pedidos ao *ClassifierService* para a identificação da possibilidade de um utente contrair a patologia em análise.

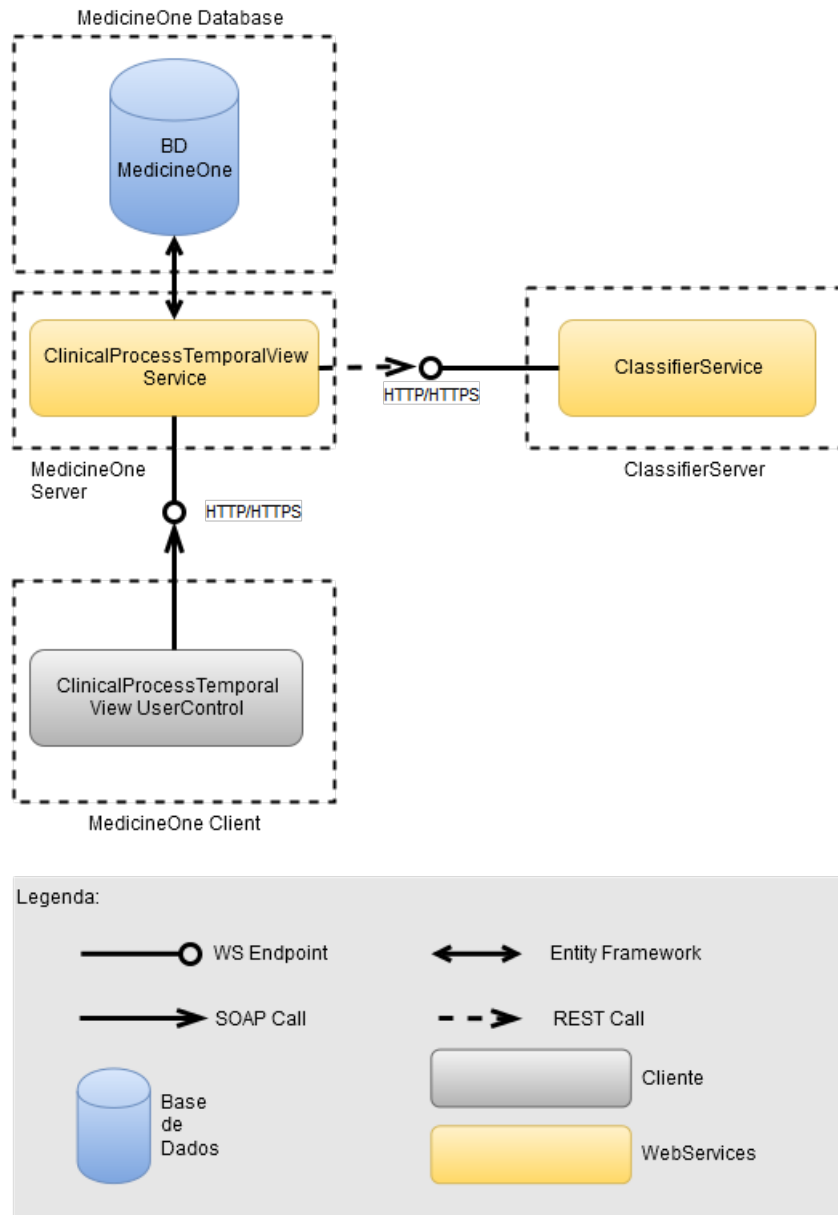


Figura 5.3: Diagrama de componentes a desenvolver no sistema

No diagrama 5.4 podemos observar que o serviço de classificação deverá ser um serviço central onde os vários servidores do *MedicineOne* devem realizar os pedidos. Sendo este serviço central, a atualização do modelo é simplificada pois só é necessária a realização da atualização numa localização e não nos várias localizações do *MedicineOne*. Esta atualização, do modelo, deve ser realizada mensalmente de forma a que o sistema de classificação utilize os dados mais recentes permitindo uma melhor classificação.

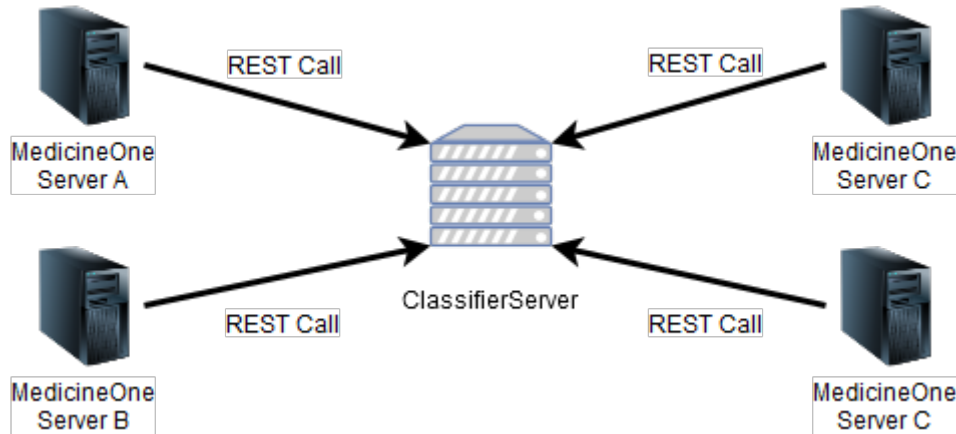


Figura 5.4: Diagrama de *deployment* do serviço de classificação.

5.1.3 Arquitetura lógica

Na arquitetura lógica do nosso sistema é possível visualizar a separação da lógica do sistema pelas diferentes camadas (*Data*, *Business* e *Presentation*). Também podemos observar os pontos transversais à aplicação que estão presentes nas várias camadas.

O modelo apresentado para a arquitetura lógica do sistema é inspirado no guia arquitetural *Microsoft Patterns and Practices* para aplicações empresariais em que se visa a aplicação dos pontos a seguir apresentados:

- **Separação dos componentes de UI dos componentes de processo.** Com este ponto pretende-se que os componentes de processo possam ser utilizados por várias interfaces do utilizador.
- ***Service Interface Contracts* para a interação com os serviços.** Com os contratos de interface pretende-se reforçar a interoperabilidade dos serviços pelos vários clientes dos serviços.
- ***Entity Management*.** Os serviços utilizam componentes de *Data Access* e agentes para obtenção e manipulação dos dados. Por exemplo "*PatientService*" utiliza um componente de *Data Access* para a obtenção da lista de pacientes da base de dados e o preenchimento de uma lista de entidades com os dados obtidos. Sendo por fim esta lista é devolvida e ligada aos componentes presentes na *user interface*.
- ***Workflows Management*.** Processos complexos podem ser orquestrados através de fluxos, de forma a simplificar a execução dos mesmos. Os fluxos são

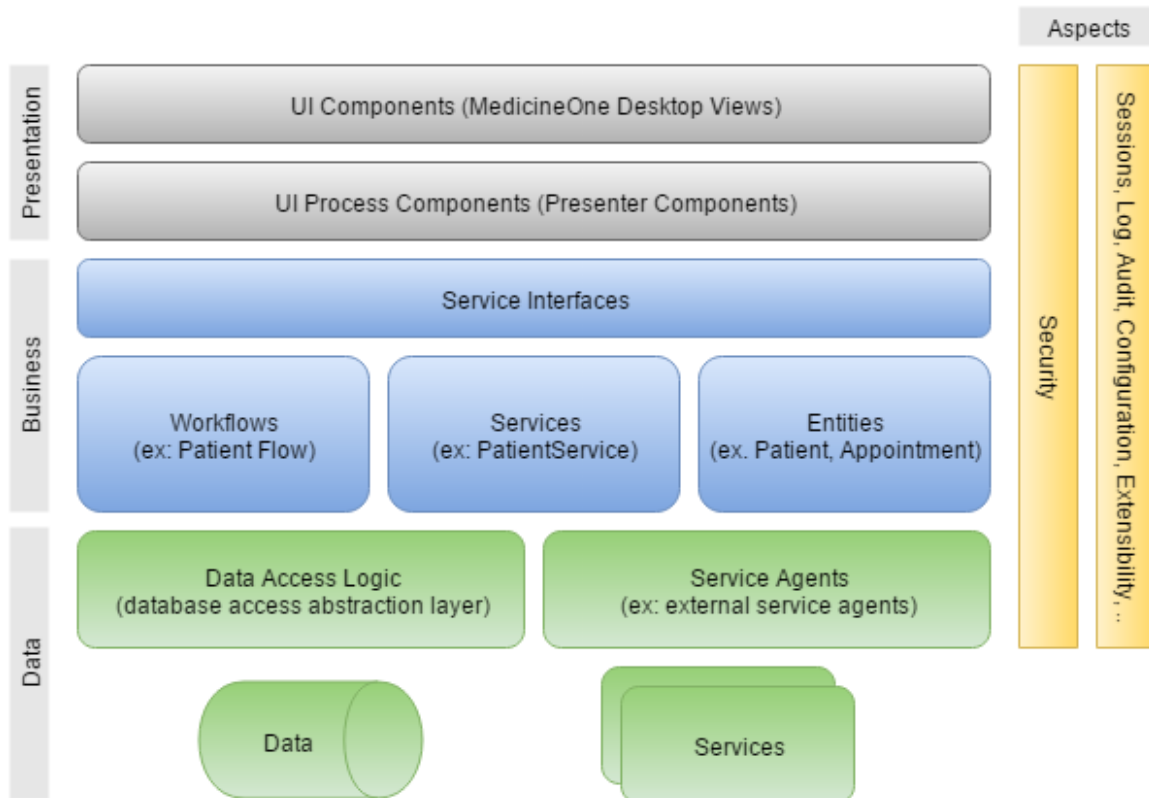


Figura 5.5: Arquitetura lógica

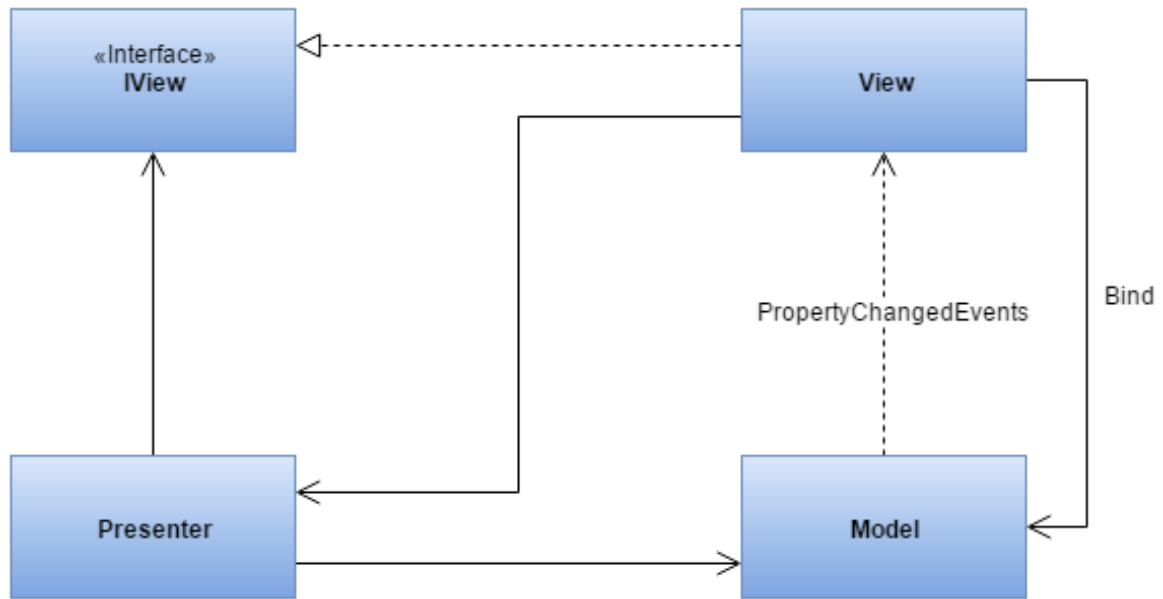
constituídos por atividades com um conjunto de propriedades. Por exemplo: um *PatientService* cria uma instância de um fluxo (*PatientAttendance*), que define um processo envolvido no atendimento de um paciente.

- **Aspect oriented approach to cross-cutting concerns.** Aspectos transversais à aplicação, como por exemplo autorizações de acesso, *logging*, segurança devem ser tratados de uma forma consistente e a sua implementação deve ser separada da lógica de negócio.

5.1.4 Model View Presenter

Para o desenvolvimento do componente a integrar no *MedicineOne 8* decidiu-se aplicar o *design pattern Model View Presenter* (MVP), à semelhança dos outros componentes da aplicação que também seguem este *design pattern*. Foi aplicado a variante deste *design pattern*, *MVP Supervising Controller*, uma vez que se pretendia que o componente seja agnóstico para que na realização de alterações à *View* não seja necessária a alteração do código existente no *Presenter*.

- **Model** - Responsável pelo acesso e manipulação dos dados. Neste caso os objetos de negócio presentes no *Model* são os serviços e repositórios de informação.
- **View/IView** - Na *View* encontram-se todos os objetos de UI, sendo que esta apenas implementa uma interface onde é definido o contrato de UI. A *View* mostra os dados que foram obtidos pelo *Presenter* e delega os eventos ao *Presenter* de forma a este efetuar o seu processamento.

Figura 5.6: *Model View Presenter*

- **Presenter** - Ao *Presenter* é aplicado o padrão de *Controller*, sendo este responsável pela orquestração dos processos de gestão de dados e dos eventos do sistema. Para além destes aspetos o *Presenter* mostra a informação na *View* e pode processar eventos relativos ao *Model*.

5.2 Estruturas de dados criadas

Nas imagens seguintes são apresentadas os diagramas de classes referente aos DTO (*Data Transfer Object*) para a transferência de dados entre o servidor e o cliente. As estruturas criadas têm por base as necessidades do cliente para que o processamento da informação seja feito maioritariamente do lado do servidor. Como cada um dos três momentos tem as suas necessidades específicas foram criados os respetivos DTO's como são apresentados nas figuras 5.7, 5.8 e 5.9, apresentadas abaixo.

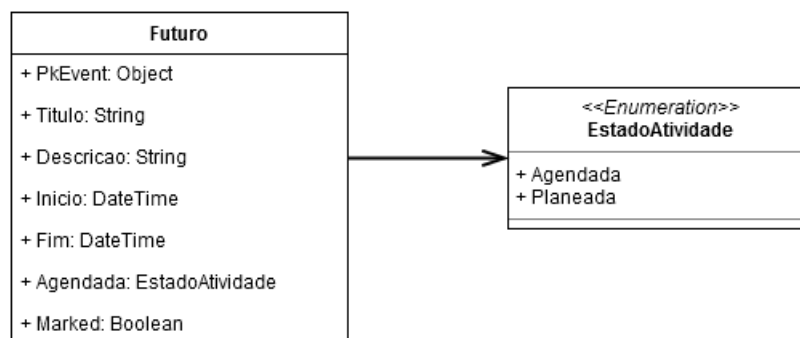


Figura 5.7: DTO relativos ao Futuro

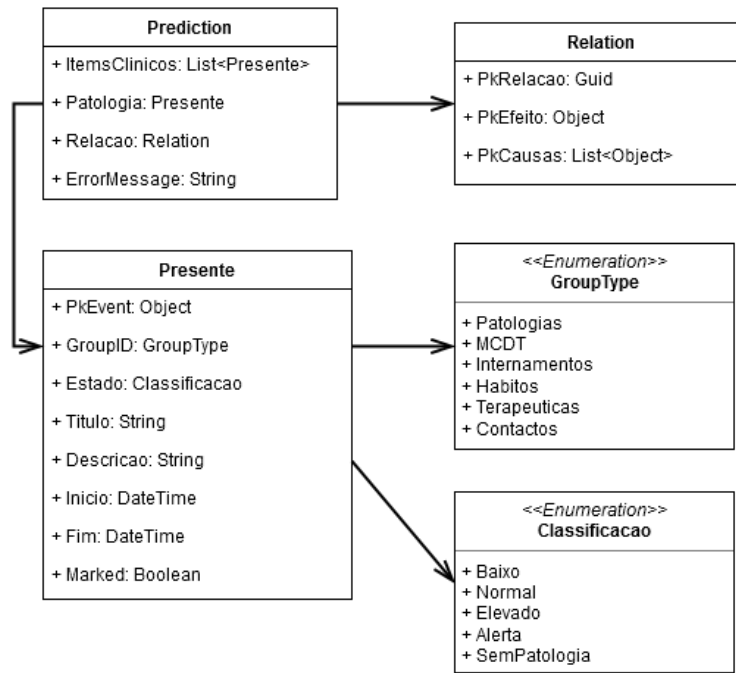


Figura 5.8: DTO relativos ao Presente

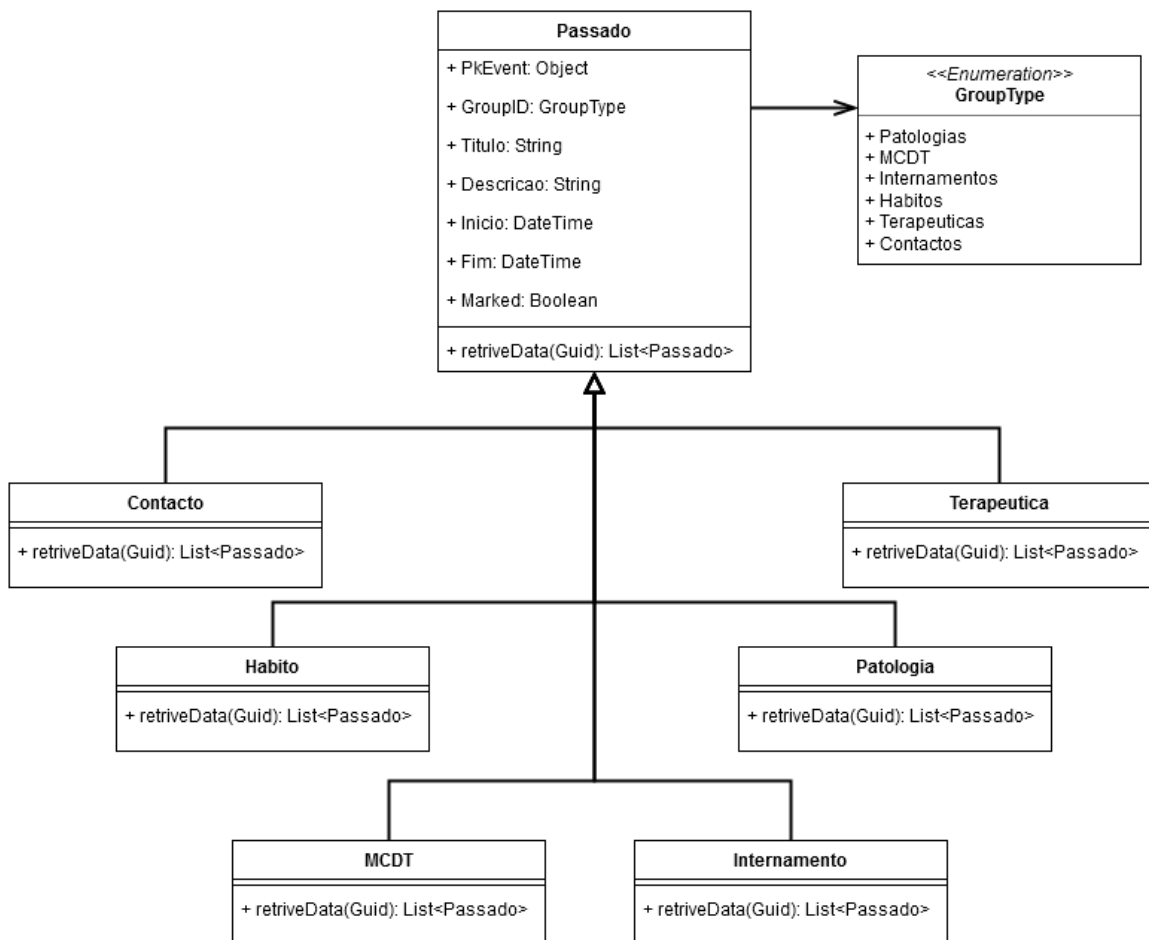


Figura 5.9: DTO relativos ao Passado

Relativamente a este último diagrama(fig. 5.9), tem-se a acrescentar que as res-

petivas classes derivadas têm como principal objetivo a recolha dos dados referentes a cada umas das categorias representadas pelas mesmas. Com isto pretendemos ter modularidade no sistema de forma a que a adição de uma nova categoria de dados seja simples sem grandes necessidades de alterações ao código.

5.3 Tecnologias

Para o desenvolvimento deste projeto serão utilizadas as tecnologias apresentadas a seguir. Estas tecnologias foram selecionadas tendo em conta vários aspetos relativos ao sistema onde o componente será integrado, a arquitetura definida e os testes realizados às mesmas.

- ***.NET Framework 4.0***

- **C#** - É utilizado C# como linguagem de programação uma vez que o desenvolvimento do *MedicineOne 8* utiliza esta linguagem.
- **WCF** - Para o desenvolvimento dos serviços foi selecionada a tecnologia *Windows Communication Foundation*, uma vez que esta se encontra presente na *framework* utilizada e os outros serviços do sistema também já utilizam esta tecnologia. O protocolo de comunicação utilizado com este serviço é SOAP à semelhança dos que atualmente já existem implementados.
- **Entity Framework 6.1** - Esta *framework* é responsável pela gestão das entidades e o seu mapeamento nas respetivas tabelas da base de dados.

5.3.1 Desenvolvimento da linha do tempo

Para o desenvolvimento da linha do tempo foi utilizado um pacote de componentes de UI

- **DevExpress WinForms Controls** - Para os controlos de UI foi selecionado o pacote da *DevExpress*, um vez que este já era utilizado no *MedicineOne 8* e pertencia às *guidelines* da empresa para o desenvolvimento de interfaces. Embora tenha sido selecionado este pacote de controlos de UI para o desenvolvimento da linha do tempo, em si, podiam terem sido utilizados controlos de outros pacotes existentes, mas no decorrer da realização dos testes para a escolha do componente a utilizar para a linha do tempo verificou-se que este pacote era o que trazia mais vantagens em relação aos outros em teste (Seção 6.1 - Suite de testes).

5.3.2 Detecção de eventos futuros no processo clínico

Para a realização da segunda parte do estágio, para realizar a detecção de eventos futuros no processo clínico, nomeadamente a detecção se um utente pode contrair diabetes foram utilizadas as ferramentas seguintes:

- *Python*

- *Orange Machine Learning*¹ - É uma *framework* que pode ser utilizada para *Data Mining* assim como *Machine Learning*. Esta *framework* utiliza as ferramentas de *machine learning* da biblioteca *Sklearn*². Permitindo, assim, uma utilização dos mecanismos através de *workflows* interativos.
- *Flask*³ - Para o serviço que irá efetuar a classificação das instâncias utilizando o modelo gerado foi selecionada a tecnologia *Flask* que nos permite a criação de um serviço em *python*, uma vez que a ferramenta de *machine learning* é uma ferramenta em *python*.

¹<http://orange.biolab.si/>

²<http://scikit-learn.org/>

³<http://flask.pocoo.org/>

Capítulo 6

Trabalho Desenvolvido

No decorrer do primeiro semestre da realização deste projeto de estágio foram realizadas várias tarefas. Estas tarefas passaram numa fase inicial pelo estudo do estado da arte sobre o tema a abordar e o objetivo pretendido para o projeto, em que serviu para conhecer o que já existia sobre este tema e para complementar as ideias do projeto de forma a este ficar mais completo, oferecendo uma melhor experiência ao utilizador final.

Após a realização deste estudo inicial iniciou-se a análise dos requisitos do projeto. Esta análise partiu da especificação do projeto apresentada pela empresa, em conjunto com os *mockups* fornecidos inicialmente e pela análise das ideias que foram surgindo durante o estudo do estado da arte. No decorrer desta fase do projeto os *mockups* iniciais necessitaram de ser alterados com o objetivo de acompanharem os requisitos que estavam a ser desenvolvidos.

Ainda no decorrer da fase da análise dos requisitos foi também iniciada uma outra fase, que decorreu em paralelo, para a realização de testes sobre alguns componentes de *User Interface*. Esta fase teve como finalidade verificar quais os componentes que possibilitavam a realização do maior número das funcionalidades pretendidas e melhor se adaptariam na aplicação onde o componente estará integrado, o *MedicineOne 8*. Para tal optou-se por realizar um pequeno projeto intitulado de Suite de Testes (apresentado no ponto 6.1) em que foram criados e testados os diversos componentes que poderiam vir a ser utilizados. Esta fase estendeu-se até ao início do desenvolvimento uma vez que o estagiário tinha pouca experiência com as tecnologias utilizadas no decorrer do projeto, podendo assim aproveitar esta oportunidade para se ambientar às tecnologias que iria utilizar na fase de desenvolvimento.

Após o término da fase da análise de requisitos foi iniciado o estudo da arquitetura do módulo a desenvolver e para tal foi necessário o estagiário conhecer o funcionamento do sistema onde será integrado o trabalho desenvolvido. Após a análise da arquitetura foi efetuado o desenvolvimento e especificação da arquitetura para o módulo a implementar e os modelos de dados auxiliares a utilizar. Uma vez que os dados já se encontram armazenados com um modelo existente, apenas necessária a sua reestruturação para os apresentar.

6.1 Suite de testes

A aplicação de teste desenvolvida teve como objetivo a experimentação de diversos controlos de *User Interface*(UI) para verificar a possibilidade de personalização e as funcionalidades que estes disponibilizam, para que permitam ao máximo cumprir os requisitos acordados com o cliente.

Para a experimentação dos vários componentes foram definidos alguns requisitos/objetivos que os componentes tinham que cumprir. Estes objetivos tiveram em conta os requisitos definidos para o projeto. A lista seguinte apresenta os objetivos delineados que os componentes tinham que cumprir para que seja possível cumprir os requisitos do projeto.

1. Criação de várias faixas horizontais para a disposição dos dados.
2. Criação de um *overview* da linha do tempo.
3. Disposição de eventos com recurso a diferentes cores.
4. Capacidade para realizar *zoom*.
5. Personalização dos *layouts* de forma a ficarem o mais parecidos possíveis com os *mockups*.
6. Representação de relacionamento entre vários eventos.

Tendo em conta os objetivos acima listados os componentes foram testados até que se verificasse que algum dos objetivos não era possível de ser cumprido ou a sua realização tivesse um grau de complexidade muito elevado.

6.1.1 *Silverlight & WPF Timeline Control*

O componente *Silverlight & WPF Timeline Control*¹ foi desenvolvido para a *framework Silverlight* e também como um componente WPF(*Windows Presentation Foundation*). O seu desenvolvimento teve por base o projeto *opensource SIMILE Timeline*².

Para a realização dos testes para este projeto utilizou-se o componente da versão WPF, uma vez que a *framework Silverlight* encontra-se *deprecated* e já foi anunciado, pela Microsoft, o fim do suporte da mesma em 2021. Na figura 6.1 é possível observar o resultado da experimentação realizada sobre este componente.

Durante a fase em que este componente foi testado foi possível concretizar alguns dos pontos em avaliação, mas foram encontrados alguns problemas que vieram a impossibilitar a concretização de alguns dos objetivos. As principais metas que não foram possíveis de atingir foram os objetivos 5 e 6, definidos no ponto anterior. Quanto ao quinto objetivo o facto de mudar o fundo do componente não é possível sem que seja necessário a reconstrução completa do mesmo. Quanto ao outro objetivo não foi encontrada uma forma de representar as relações entre os eventos. Relativamente aos restantes objetivos foi possível a sua conclusão.

¹<https://timeline.codeplex.com/>

²<http://www.simile-widgets.org/timeline/>

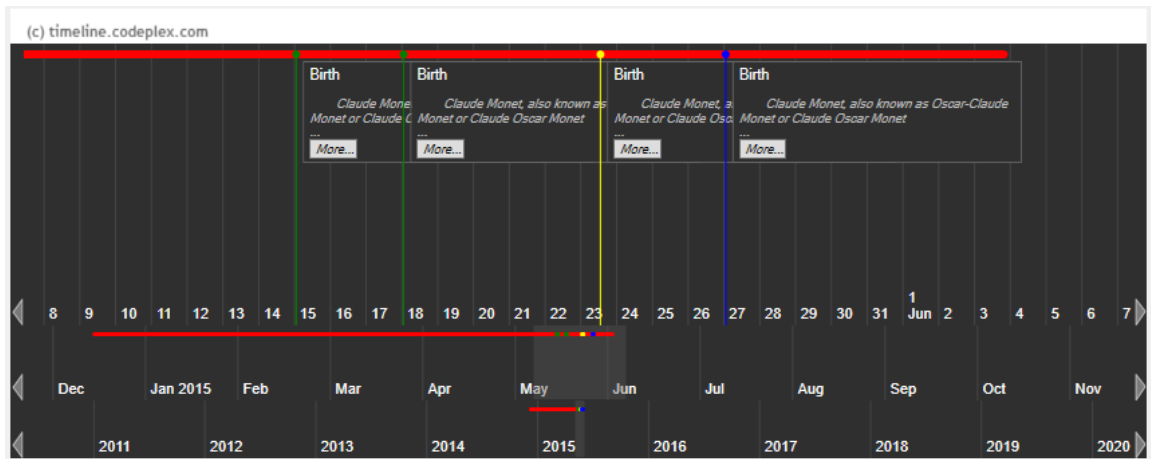


Figura 6.1: *SIMILE WPF Timeline Control*

6.1.2 *WPF Scheduler Timeline Control*

O componente *WPF Scheduler Timeline Control*³ está incluído nos pacotes de componentes UI da *Infragistics*⁴. Para a experimentação deste componente foi utilizada a versão *trial* do mesmo sendo que se tentou cumprir os requisitos acima descritos. Na figura 6.2, é possível observar o resultado obtido após a conclusão dos testes.

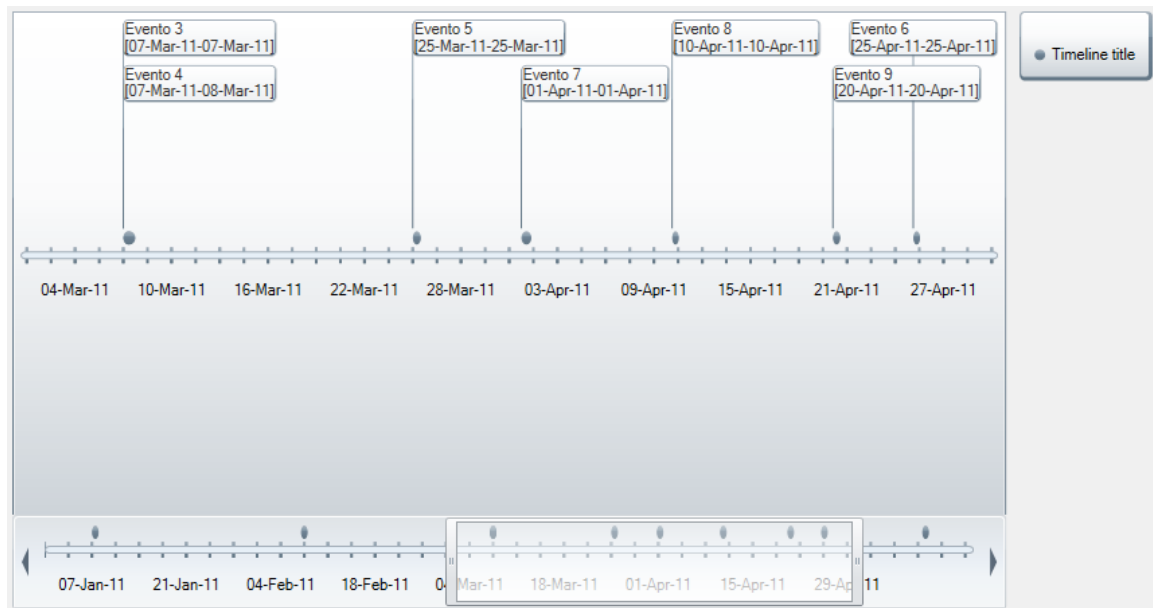


Figura 6.2: *WPF Infragistics Timeline Control*

Neste componente não foi possível criar diferentes faixas para dispor os eventos, sendo que este facto impossibilita a utilização deste componente para a visualização do "Presente" uma vez que neste são precisas múltiplas faixas de forma a poder representar os valores normais e anormais. Para além da criação das faixas também não foi possível efetuar a personalização do *layout* como o pretendido e não sendo também possível representar relacionamentos entre múltiplos eventos.

³<http://www.infragistics.com/products/wpf/schedules/timeline>

⁴<http://www.infragistics.com/>

6.1.3 Winforms Scheduler Control + Range Control

Neste ponto são apresentados duas experiências realizadas com componentes da *DevExpress*. Aplicou-se um esforço extra no teste destes componentes uma vez que a empresa já utiliza estes componentes na sua solução, embora no início do estágio ainda estava a utilizar a versão *DevExpress 11.2*, e no decorrer do estágio estar prevista o seu *upgrade* para a versão *DevExpress 15.2*. Como já estava prevista a realização do *upgrade*, os testes foram realizados sobre a versão *DevExpress 15.2*, para retirar proveito de novas *features* que tenham surgido posteriormente à versão *DevExpress 11.2*. Estes componentes utilizados pertencem ao conjunto de componentes do tipo *WinForms* disponibilizados pela *DevExpress*.

Como o *Scheduler Control* não permite a criação de um *overview* da linha do tempo foi utilizado em conjunto com este um outro componente para a criação do respetivo *overview* da linha do tempo. Este segundo componente foi o *Range Control*⁵, que nos veio permitir a criação do *overview* da linha do tempo e efetuar as respetivas interações necessárias entre os dois componentes.

Winforms Timeline View

Na primeira experiência realizada com o componente *Scheduler Control* foi utilizada uma das suas *features*, a *Timeline View*⁶. Esta *feature* é uma forma de visualização do componente em que a informação é mostrada numa linha do tempo.

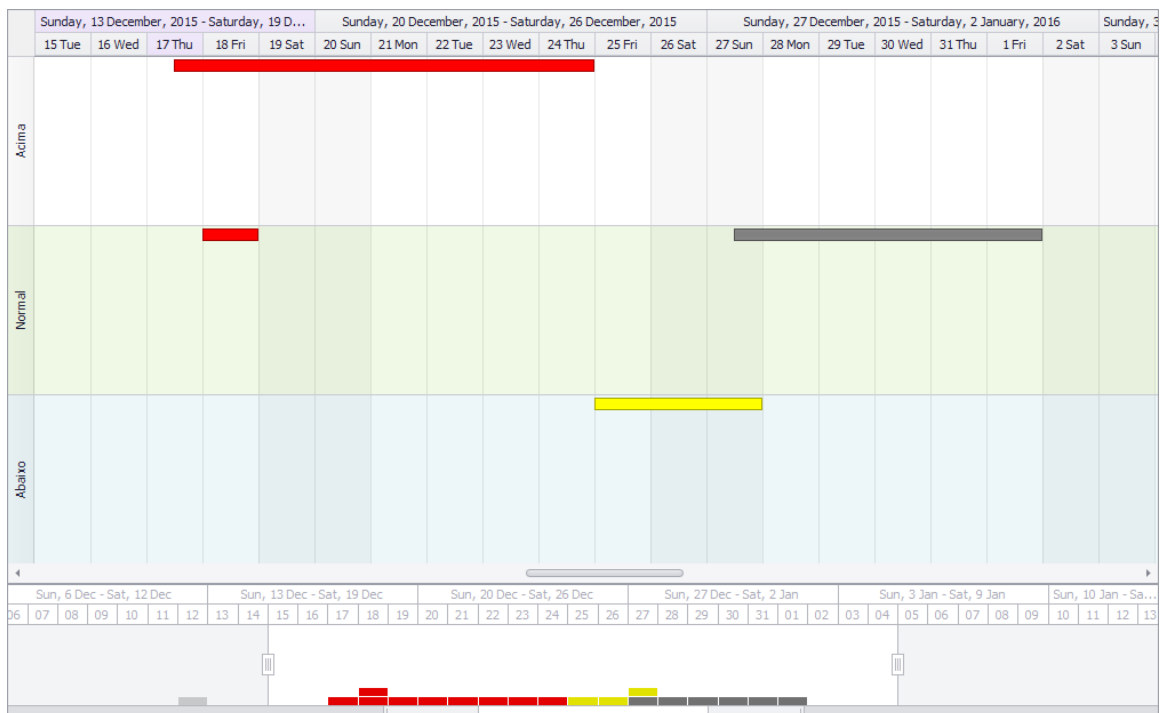


Figura 6.3: *DevExpress Winforms Scheduler Control - Timeline View*

Nos testes realizados a este componentes já foi utilizado o *Range Control*, componente que permite a criação do *overview* da linha do tempo, para serem efetuados a experimentação da interação entre os dois componentes. Na figura 6.3 podemos

⁵<https://documentation.devexpress.com/#WindowsForms/CustomDocument11739>

⁶<https://documentation.devexpress.com/#WindowsForms/CustomDocument3109>

observar o resultado final após a conclusão da experimentação deste componente em que é possível verificar os vários objetivos que este componente satisfaz.

No decorrer da experimentação foi possível observar que este era um componente promissor pois as diferentes funcionalidades que este deveria suportar eram possíveis de realizar apenas com a exceção da representação gráfica de relações entre eventos.

Gantt View

Como este componente com recurso à *Timeline View*, apresentou bons resultados na medida em que apenas não foi possível representar as relações entre eventos, decidiu-se experimentar outra vista deste componente, a *Gantt View*⁷. Optou-se pela experimentação desta vista uma vez que os diagramas de *Gantt* apresentam-nos as tarefas com recurso a uma linha do tempo e também fornecem a possibilidade de visualizar relações entre as diferentes tarefas. A figura 6.4 apresenta-nos o resultado final após a experimentação do componente.



Figura 6.4: *DevExpress Winforms Scheduler Control - Gantt View*

Na figura é possível observar que este componente permite executar os diferentes requisitos definidos no início deste tópico. Sendo que é possível criar diferentes faixas, um *overview* da linha do tempo com recurso ao *Range Control*, aplicar cores aos diferentes eventos, personalizar o *layout* do componente e representar relações entre os diversos eventos. Para além destes requisitos mais visíveis também é possível

⁷<https://documentation.devexpress.com/#WindowsForms/CustomDocument10698>

realizar *zoom* sobre o componente para uma melhor visualização da informação neste contida.

6.1.4 Conclusões

Na tabela 6.1 é possível observar de uma forma resumida os resultados das experiências realizadas com os diferentes componentes. Em que se encontram assinalados os diferentes pontos, referentes à listagem apresentada no início do tópico, que cada componente facilita a possibilidade de realizar.

Tabela 6.1: Resultados das experiências realizadas aos componentes. (RC - *Range Control*)

	<i>SIMILE</i>	<i>Infragistics</i>	<i>DevExpress</i>	
	<i>WPF Timeline Control</i>	<i>WPF Scheduler Timeline Control</i>	<i>Scheduler Timeline View + RC</i>	<i>Scheduler Gantt View + RC</i>
1. Faixas Horizontais	✓	✗	✓	✓
2. <i>Overview</i>	✓	✓	✓	✓
3. Cores Eventos	✓	✓	✓	✓
4. Zoom	✓	✓	✓	✓
5. Personalização <i>Layout</i>	✗	✗	✓	✓
6. Relações entre Eventos	✗	✗	✗	✓

Na tabela acima é possível observar que apenas o componente da *DevExpress Scheduler Control - Gantt View* em conjunto com o *Range Control*, permite-nos efetuar os diferentes requisitos propostos pelo cliente para o projeto. Em relação aos outros componentes testados, estes na fase de experimentação falharam em alguns pontos importantes o que fez com que estes ficassem inviabilizados para o projeto. Alguns dos pontos foram considerados como impossíveis de realizar, pelo facto de que para a realização dos mesmos ser necessário um maior esforço aquando comparados com os outros componentes em teste.

6.2 Protótipo

Neste ponto é apresentado o trabalho desenvolvido relativamente à implementação da linha do tempo. Sendo que devido às alterações efetuadas ao plano inicial, relativamente à segunda fase deste estágio, não foi possível terminar a linha do tempo, optou-se por não concluir alguns requisitos que não trariam uma maior complexidade ao trabalho uma vez que estes eram semelhantes a outros já implementados e que para serem implementados poderiam demorar muito tempo que iria atrasar os avanços da segunda parte ou até mesmo comprometer a mesma.

6.2.1 Passado

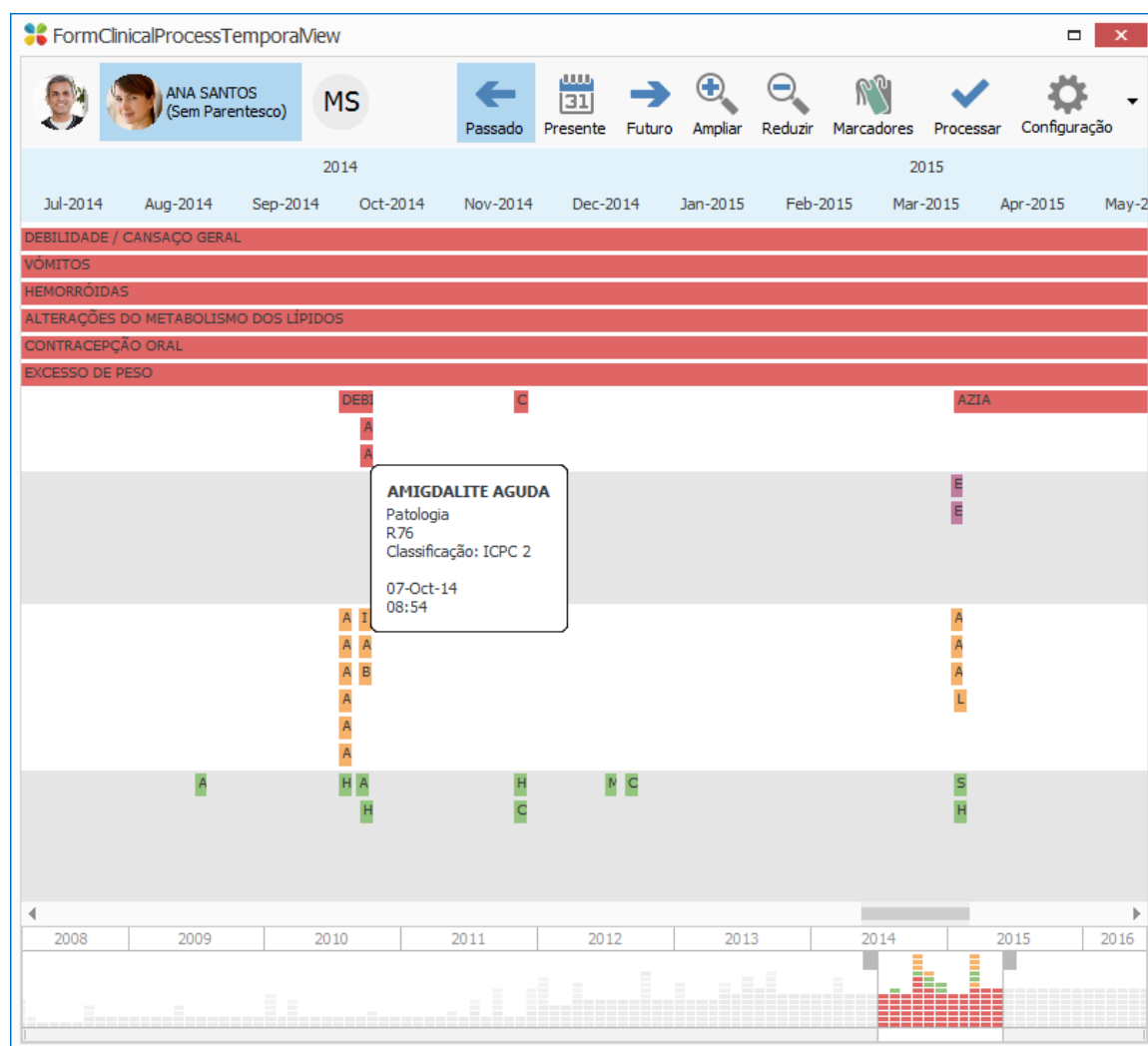


Figura 6.5: Ecrã final relativamente ao momento do Passado.

6.2.2 Presente

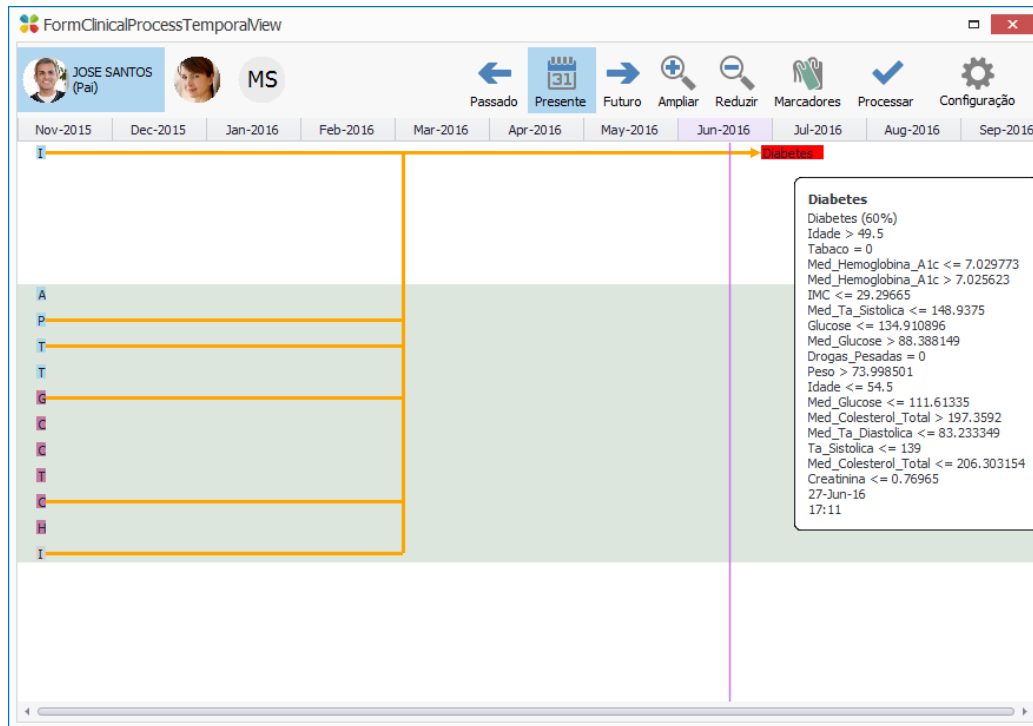


Figura 6.6: Ecrã final relativamente ao momento do Presente.

6.2.3 Futuro

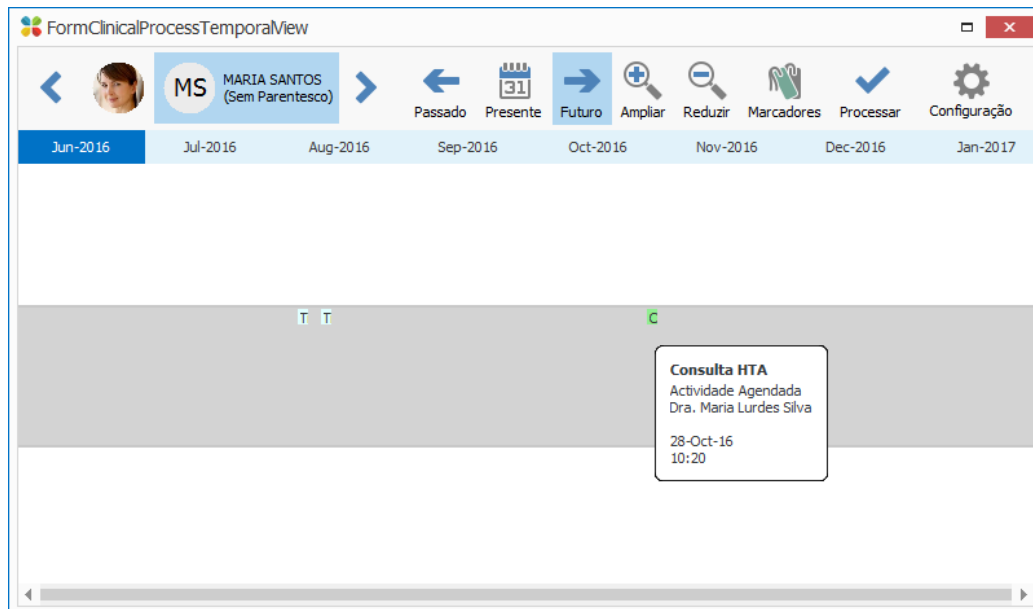


Figura 6.7: Ecrã final relativamente ao momento do Futuro.

6.2.4 Configuração

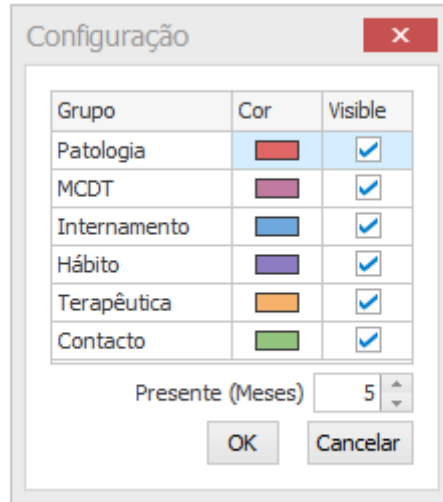


Figura 6.8: Ecrã final do menu de configuração.

6.3 Detecção de eventos futuros no processo clínico

Relativamente à segunda parte deste estágio, foi decidido efetuar uma abordagem de reconhecimento de padrões para criar um sistema que permita detetar se um determinado utente pode ter um problema de saúde e o que poderá estar na sua origem. Como isto pode ser um sistema muito complexo decidiu-se inicialmente simplificar o problema e efetuar a deteção se um paciente pode vir ou não a contrair diabetes, uma vez que este é um problema de saúde muito comum na sociedade.

Para os testes realizados foi necessária a criação de um *dataset* com dados reais anonimizados, que nos permitissem detetar os possíveis padrões na vida dos utentes de forma a perceber a possibilidade destes poderem contrair diabetes. Para a criação do *dataset* foi necessário efetuar a implementação de uma aplicação para efetuar a recolha dos dados da base de dados e a sua transformação para depois serem utilizados.

6.3.1 Características

Para o *dataset* foram consideradas algumas características consideradas relevantes para o problema, sendo elas:

- **Tabaco** - Característica discreta que identifica se o utente é, ou não, consumidor de tabaco.
- **Álcool** - Característica discreta que identifica se o utente tem o hábito de consumir álcool.
- **Drogas Leves** - Característica discreta que identifica o consumo de drogas leves por parte do utente.

- **Drogas Pesadas** - Característica discreta que identifica o consumo de drogas pesadas por parte do utente.
- **Medicamentos** - Característica discreta que identifica o consumo habitual de medicamentos por parte do utente.
- **Exercício Físico** - Característica discreta que indica se um determinado utente tem o hábito de fazer exercício físico.
- **Sedentarismo** - Característica discreta que indica se o utente tem uma vida sedentária.
- **Género** - Característica discreta com o valor 0 - Feminino; 1 - Masculino.
- **Idade** - Idade do utente aquando do último registo recolhido.
- **Altura** - Esta característica representa a altura do utente no último registo da mesma.
- **Peso** - Peso do utente no último registo.
- **IMC** - Índice de Massa Corporal.
- **Sistólica** - Valor do último registo da tensão arterial sistólica.
- **Media Sistólica** - Valor médio da tensão arterial sistólica.
- **Diastólica** - Valor do último registo da tensão arterial distólica.
- **Média Diastólica** - Valor média da tensão arterial distólica.
- **Glucose** - Valor do último registo dos níveis de glucose.
- **Média Glucose** - Valor médio dos níveis de glucose.
- **Colesterol HDL** - Valor do último registo do nível de colesterol HDL(*High Density Lipoprotein*)
- **Média Colesterol HDL** - Valor médio do colesterol HDL registado no processo do utente.
- **Colesterol LDL** - Valor do último registo do nível de colesterol LDL(*Low Density Lipoprotein*)
- **Média Colesterol LDL** - Valor médio do colesterol LDL registado no processo do utente.
- **Colesterol Total** - Valor do último registo do nível de colesterol total.
- **Média Colesterol Total** - Média de todos os registos dos níveis de colesterol total.
- **Hemoglobina A1c** - Valor do último registo do nível da hemoglobina glicada.
- **Média Hemoglobina A1c** - Média de todos os registos dos níveis da hemoglobina glicada.

- **Triglicérides** - Valor do último registo do nível dos triglicérides.
- **Média Triglicérides** - Média de todos os registos dos níveis dos triglicérides.
- **Creatinina** - Valor do último registo do nível de creatinina.
- **Média Creatinina** - Média dos níveis de creatinina registados.
- **Hemoglobina** - Valor do último registo da hemoglobina
- **Média Hemoglobina** - Média de todos os registos da hemoglobina.

Foram criados três *datasets*, para treino, validação e teste, com as características acima identificadas sendo que cada instância corresponde aos dados clínicos de um utente. O *dataset* recolhido para treino contém cerca de **308380** instâncias, sendo que a sua distribuição por idade, género e classe pode ser visualizada nos gráficos seguintes (fig. 6.9, 6.10 e 6.11). Para além deste *dataset* também foram recolhidos mais dois *datasets* independentes de dimensões menores, os quais serviram para teste, com cerca de 3600 instâncias, e validação do modelo com aproximadamente 34780 instâncias. Os dados recolhidos, nas instâncias em que eram identificadas como contendo diabetes, são dados referentes a registos anteriores à deteção da patologia, isto deve-se ao facto de o objetivo ser a previsão da possibilidade de um utente contrair diabetes.

- Gráfico distribuição de idades

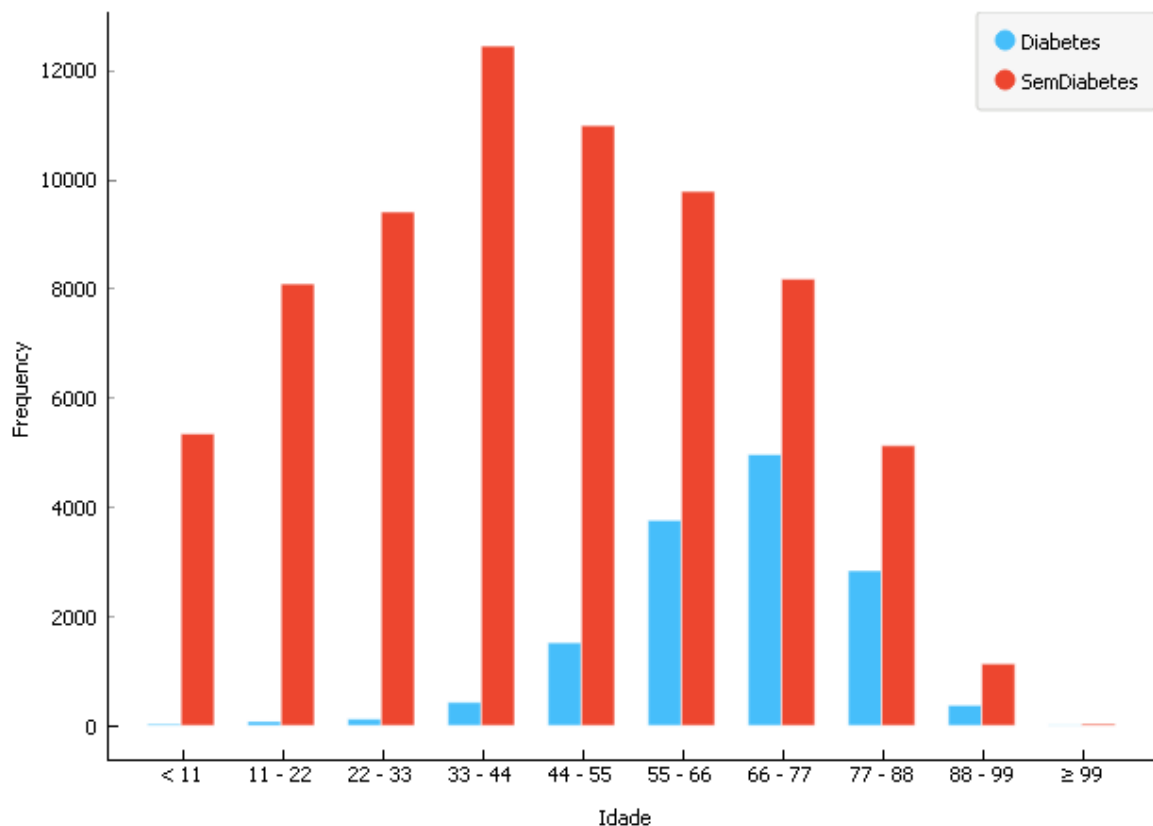


Figura 6.9: Distribuição de idades por classe.

- Gráfico distribuição do género

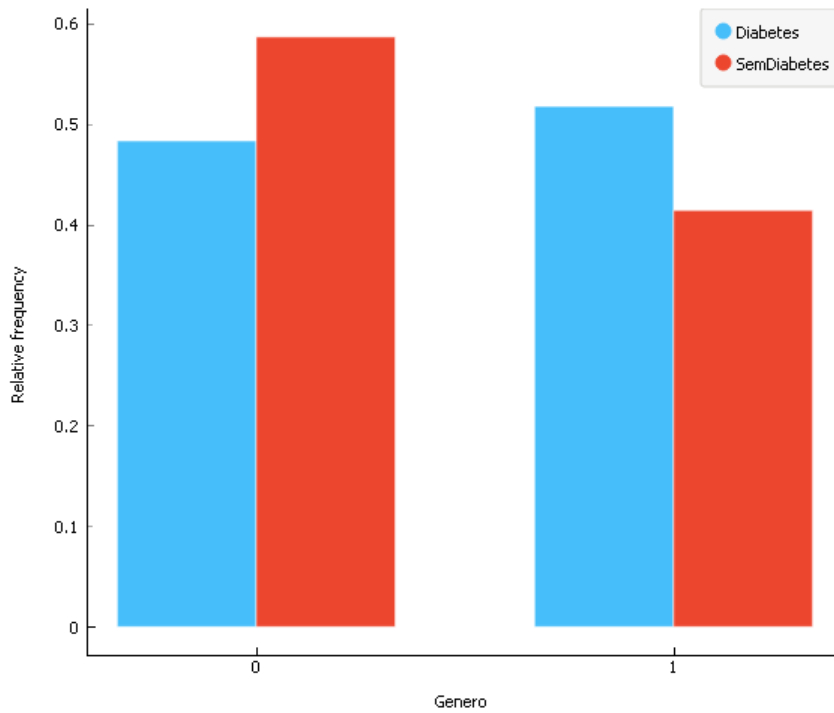


Figura 6.10: Frequência relativa da distribuição de género pelas classes(0 - Feminino; 1-Masculino).

- Gráfico distribuição das classes

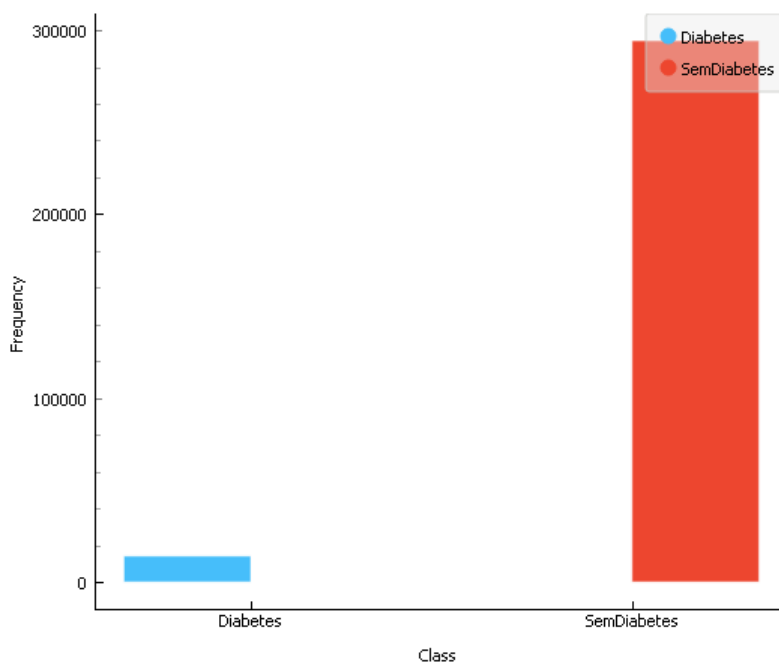


Figura 6.11: Distribuição dos registos por classes.

Pelo gráfico 6.11 podemos concluir que o *dataset* se encontra gravemente desbalanceado e a utilização de um *dataset* com estas características pode levar a que o modelo apenas identifique as instâncias da classe majoritariamente observada. Com isto foi necessário aplicar técnicas para balancear o *dataset* como será apresentado mais à frente no ponto 6.3.3.

6.3.2 *Missing Values*

Como nem todos os utentes realizam os mesmos exames, existiram alguns valores em falta nas *features*, sendo que foi necessário efetuar o seu tratamento de forma a que estes não tenham grande relevância no sistema, o que pode aumentar o erro do mesmo. De seguida são apresentadas as abordagens testadas para o tratamento dos valores em falta. Inicialmente foram testadas abordagens, mais simples, lineares e depois foi-se avançando para abordagens mais complexas, não lineares, terminando com a experimentação do algoritmo KNN. Para a obtenção dos gráficos seguintes recorreu-se ao *software Weka*⁸.

- **Média** - Inicialmente foi utilizado valor médio para substituição dos valores em falta. Este método não se verificou muito efetivo uma vez que todas as instâncias, que tinham um determinado atributo em falta, ficavam com o mesmo valor (fig. 6.12).

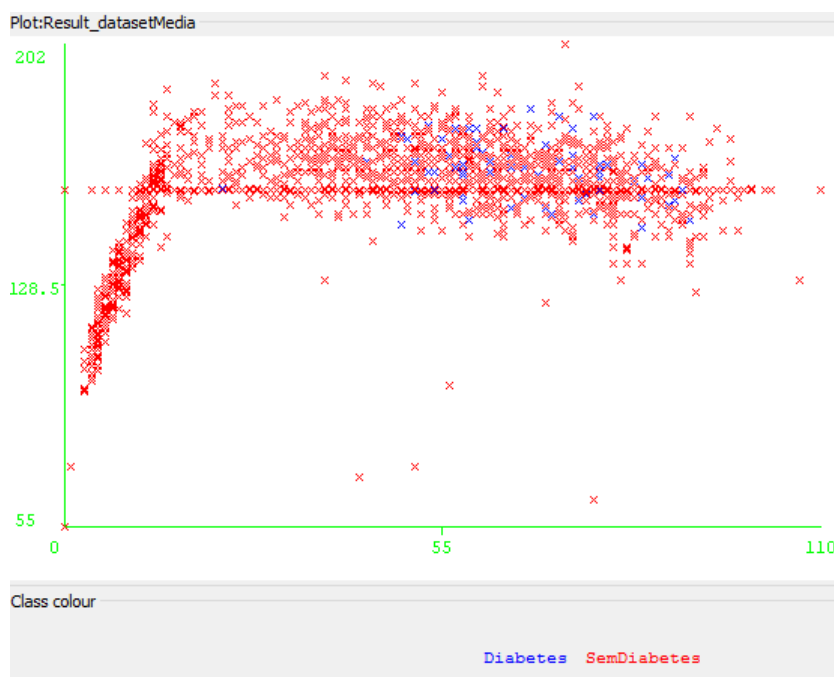


Figura 6.12: Resultado do tratamento dos *missing values* recorrendo à média. No gráfico os dados apresentados são relativos à altura por idades.

- **Mediana** - Relativamente à mediana (fig. 6.13) também não foi uma boa métrica à semelhança da média visto que a mediana nos dá o valor do ponto central de um conjunto de dados.

⁸<http://www.cs.waikato.ac.nz/ml/weka/>

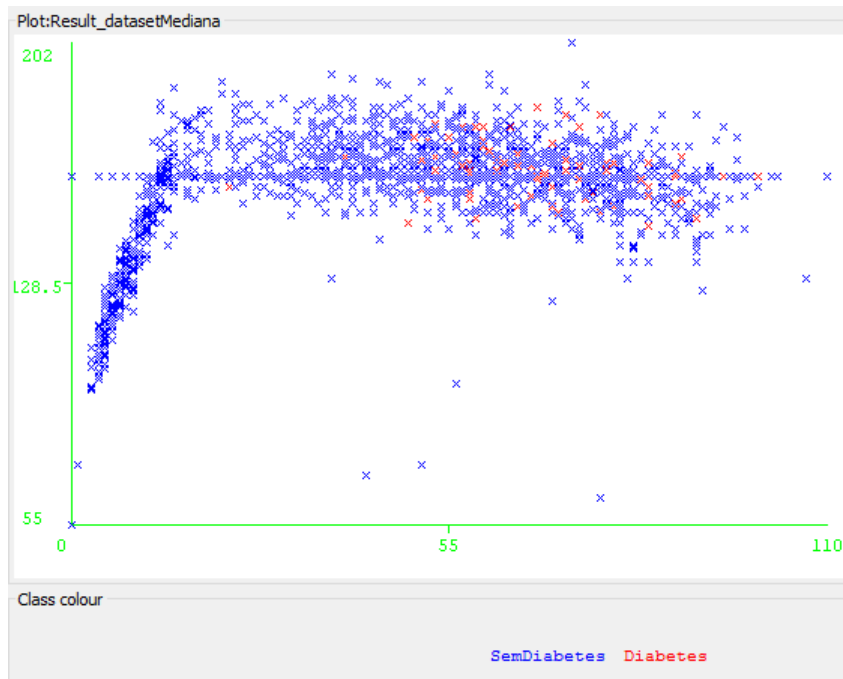


Figura 6.13: Resultado do tratamento dos *missing values* recorrendo à mediana. No gráfico os dados apresentados são relativos à altura por idades.

- **Regressão Linear** - Dado que a média e a mediana nos davam resultados muito semelhantes, foi tomada a decisão de se utilizar uma fórmula matemática a fim de se aproximar os valores imputados dos valores reais. Começou-se com uma regressão linear que por sua vez, à semelhança dos dois métodos anteriores, também não se aproximava dos valores reais (fig. 6.14), pois também é um método linear e os dados obtidos são não lineares.

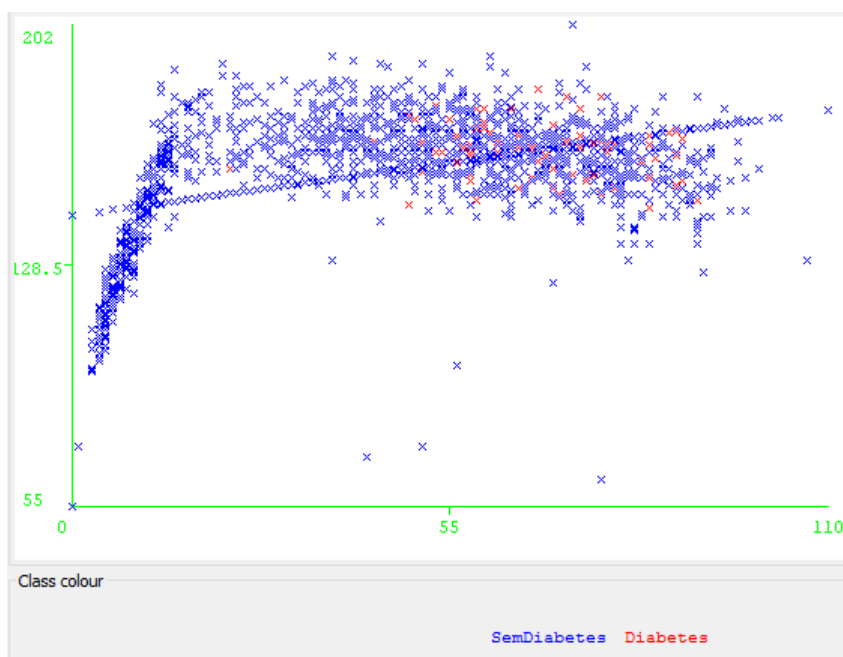


Figura 6.14: Resultado do tratamento dos *missing values* recorrendo a uma regressão linear. No gráfico os dados apresentados são relativos à altura por idades.

- **Função Quadrática** - Após a aplicação dos métodos lineares anteriores, optou-se por experimentar uma aproximação dos valores através de funções não lineares. Em primeiro lugar experimentou-se uma função quadrática (fig. 6.15) para aproximar os resultados aos valores reais. Com esta função obtiveram-se resultados melhores que os anteriores, não sendo, contudo, os ideais. Como tal, foi-se aumentado o grau da função e analisando os resultados.

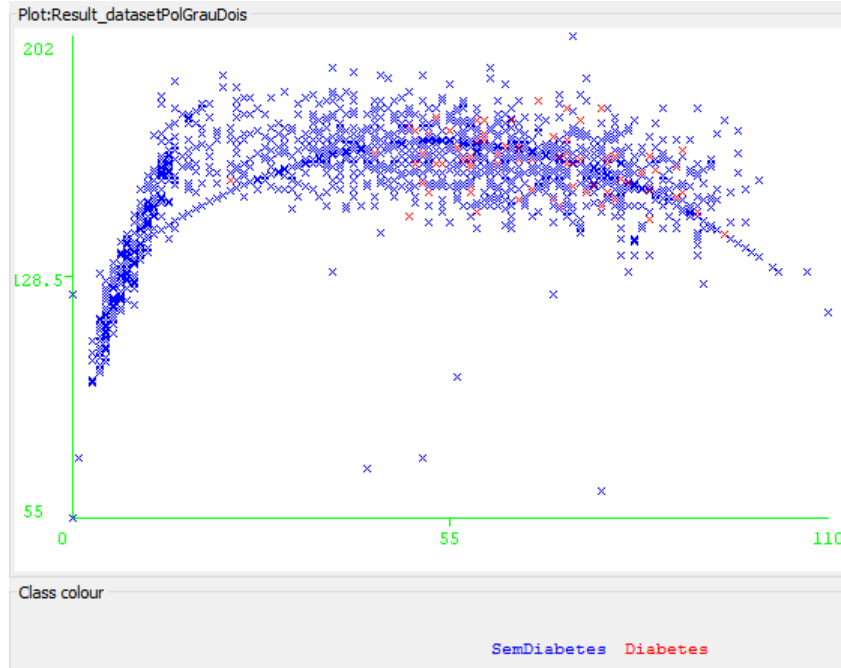


Figura 6.15: Resultado do tratamento dos *missing values* recorrendo a uma função quadrática. No gráfico os dados apresentados são relativos à altura por idades.

- **Função Cúbica** - Com recurso a uma função cúbica (fig. 6.16) já foi possível ter uma boa aproximação dos valores à realidade. Ainda assim nos últimos valores calculados ainda existia uma tendência crescente, ficando estes valores acima da realidade.
- **Polinómio de Quarto Grau** - Como as funções polinomiais estavam a dar resultados próximos do esperado, experimentou-se aumentar mais uma vez o grau da função para se obter uma melhor aproximação dos resultados à realidade, como é possível observar no gráfico da figura 6.17.
- **KNN ($k = 10$)** - Para além das funções polinomiais também foi testado o algoritmo *KNN* (*K-Nearest Neighbor*) para calcular os dados em falta. A implementação utilizada deste algoritmo foi a seguinte: eram selecionados os dez pontos mais próximos ($k = 10$) do ponto para o qual queríamos calcular o *missing value* e de seguida era feita a média dos valores desses pontos sendo esse o novo valor para o ponto em análise. Os resultados obtidos, figura 6.18, com este algoritmo encontram-se bastante próximos dos valores dos outros pontos uma vez que o algoritmo utiliza os seus vizinhos para calcular os novos valores. Neste algoritmo foi escolhido inicialmente o valor fixo 10 para testar e visualizar os dados da imputação, sendo que como estes resultados se mostraram favoráveis já não foram testados outros valores.

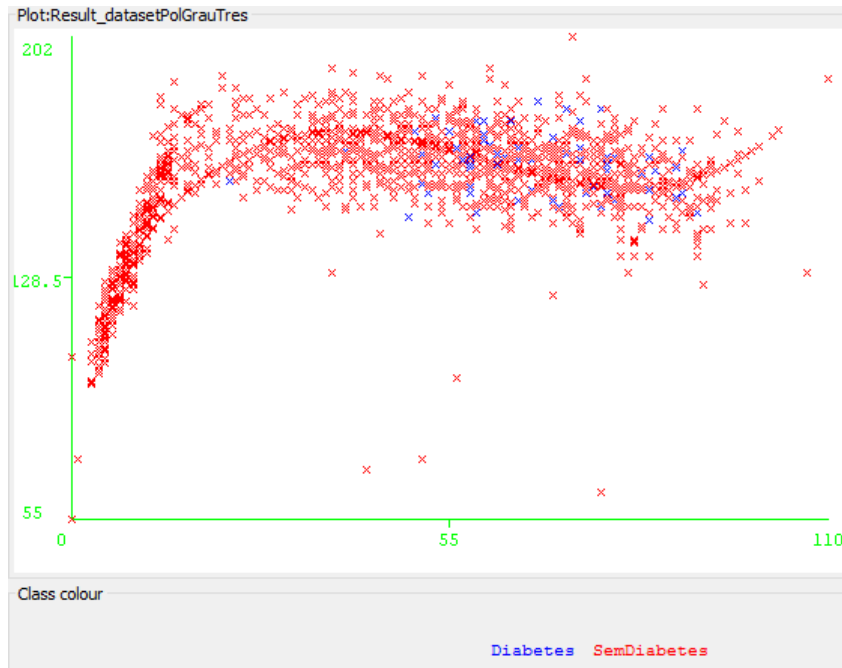


Figura 6.16: Resultado do tratamento dos *missing values* recorrendo a uma função cúbica. No gráfico os dados apresentados são relativos à altura por idades.

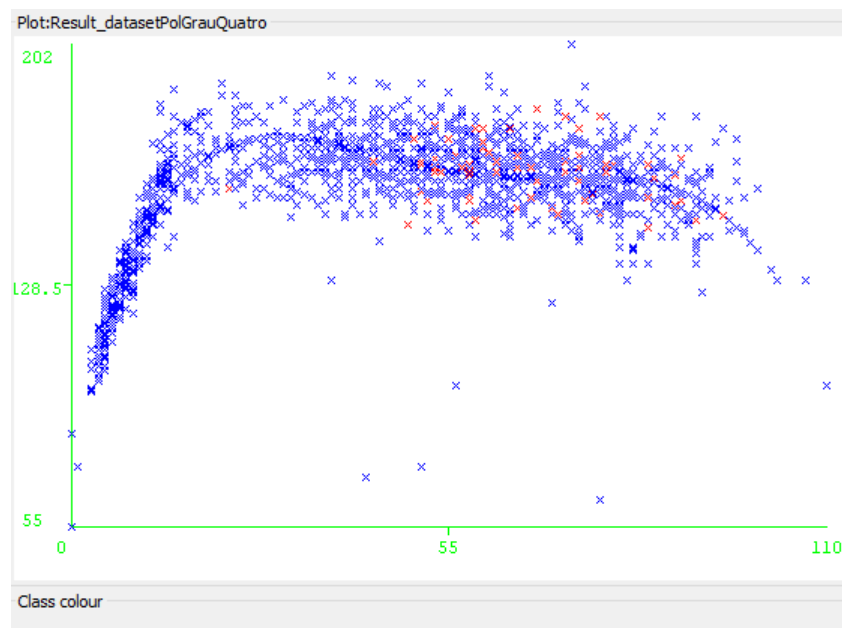


Figura 6.17: Resultado do tratamento dos *missing values* recorrendo a um polinómio de grau quatro. No gráfico os dados apresentados são relativos à altura por idades.

Para concluir, relativamente aos *missing values* foi selecionada a técnica que utiliza o algoritmo KNN uma vez que com esta técnica os resultados obtidos encontram-se perto dos resultados reais para cada ponto e não se encontram aglomerados num determinado valor como ocorre no caso das funções polinomiais.

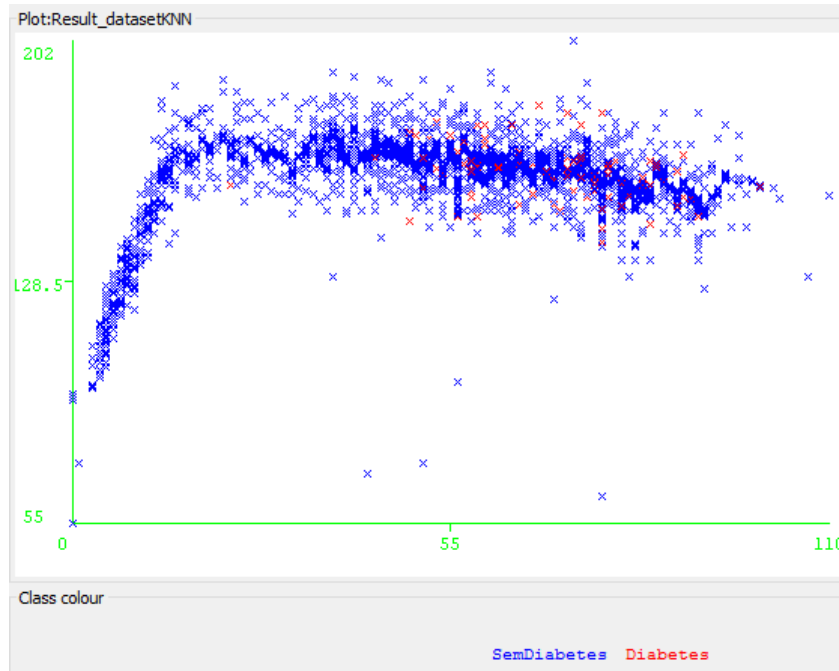


Figura 6.18: Resultado do tratamento dos *missing values* recorrendo ao método KNN. No gráfico os dados apresentados são relativos à altura por idades.

6.3.3 Data Sampling

Como existem mais pessoas que não tenham diabetes do que as que têm, o *dataset* recolhido encontra-se com um grande desequilíbrio entre as classes sendo que para obter melhores resultados foi necessária a criação de *subdatasets* para treino que estivessem mais equilibrados.

Foram aplicados vários rácios para as diferentes classes de forma a obter melhores resultados.

O *dataset* utilizado para treino tinha originalmente cerca de 294000 instâncias que correspondiam à classe "sem diabetes" e cerca de 14000 instâncias da classe "diabetes", o que perfaz um rácio original de $\approx 21/1$.

De forma a obterem-se melhores resultados foi realizado *resampling* ao *dataset* utilizando os rácios por classe listados abaixo, sendo que para cada instância da classe "diabetes" existem n instâncias da classe "sem diabetes". Para a realização dos testes foram sendo criados *datasets* com os seus rácios de 1/1 até 9/1.

Para a análise foi criado um *dataset* para cada um dos rácios e depois foi testado no sistema com os quais se obtiveram os resultados apresentados nas tabelas seguintes. A configuração do sistema utilizada na realização dos testes foi a seguinte:

- *Classification Tree*
- Variáveis contínuas
- *Information Gain (22 Features)*

Para a avaliação do sistema foram utilizadas algumas métricas para além dos resultados, essas métricas encontram-se explicadas a seguir.

- **Accuracy (ACC)** - A *accuracy* consiste nos resultados positivos do nosso sistema sobre o total de resultados. Considerar apenas esta métrica não é sufi-

ciente, uma vez que um algoritmo pode ter uma boa *accuracy* mas falhar os resultados mais importantes para o problema.

- **Area Under Curve(AUC)** - Esta métrica avalia a performance dos classificadores utilizando as curvas de ROC(*Receiver Operating Characteristic*), sendo que este valor é referente à área abaixo da curva(*Area Under Curve*).
- **F1-Score** - O *F1 Score* mede a performance de um classificador tendo em conta a *precision* e o *recall*. Apesar de ser uma métrica muito utilizada em *machine learning*, não é das melhores métricas para avaliar um classificador binário, uma vez que não tem em conta os resultados negativos classificados corretamente.
- **Precision** - Esta métrica é referente à fração de instâncias classificadas como positivas que são relevantes para o problema.
- **Recall** - O *Recall* mostra o relacionamento entre as instâncias relevantes classificadas corretamente sobre o total de instâncias relevantes.

Tabela 6.2: Métricas de performance do sistema

Rácio	AUC	ACC	F1-Score	Precision	Recall	%Correct
1 / 1	68	52.1	66.8	98.6	50.5	52.0978
2 / 1	73.1	63.4	76.5	98.8	62.4	63.4065
3 / 1	75.3	69.2	80.9	98.7	68.6	69.2137
4 / 1	73	73	83.8	98.2	73	73.0203
5 / 1	75.4	76	85.8	98.4	76	75.9933
6 / 1	76.6	78.4	87.5	98.5	78.7	78.4940
7 / 1	73.8	78.9	87.8	98.1	79.4	78.8830
8 / 1	74.7	80.6	88.8	98.1	81.1	80.5502
9 / 1	74.5	82.3	90	98	83.1	82.3006

Tabela 6.3: Instâncias identificadas corretamente

Rácio	Diabetes	SemDiabetes	%Diabetes	%SemDiabetes	#Inst
1 / 1	143	1732	85.63	50.47	28186
2 / 1	140	2142	83.83	62.41	42279
3 / 1	137	2354	82.04	68.59	56372
4 / 1	122	2506	73.05	73.02	70465
5 / 1	125	2610	74.85	76.05	84558
6 / 1	125	2700	74.85	78.67	98651
7 / 1	114	2725	68.26	79.4	112744
8 / 1	114	2785	68.26	81.15	126837
9 / 1	110	2852	65.87	83.1	140930

Com os resultados obtidos, tabelas 6.2 e 6.3, é possível concluir que quando é utilizado um rácio de 5/1 ou 6/1 obtemos resultados mais satisfatórios estando os resultados das classes mais equilibrados, o que faz com que o nosso sistema consiga diferenciar as duas classes, embora com a utilização de rácios superiores seja possível ter um maior número de instâncias totais classificadas corretamente. Isto pode não ser o resultado esperado, pois esse acréscimo no número de instâncias corretas deve-se ao facto de as classes não serem equilibradas e o que acontece é que o sistema identifica melhor as instâncias da classe mais predominante mas acaba por falhar mais instâncias da outra classe que é menos predominante.

Após a obtenção destes resultados decidiu-se avançar para os testes seguintes com os dois *datasets* com os rácios que se obtiveram os melhores resultados, sendo neste caso os *datasets* que contém os rácios 5/1 e 6/1.

6.3.4 Testes e afinação do modelo

Durante a fase de testes do classificador existiu a necessidade de efetuar algum pré-processamento aos dados de forma a melhorar a performance do classificador. Tendo em conta a obtenção de uma melhor performance foram realizados alguns testes em que foram sendo aplicadas técnicas de pré-processamento e de *feature selection*.

- **Variáveis Contínuas** - Como existiam algumas *features* que tinham valores discretos, estas foram convertidas em *features* com valores contínuos, criando uma nova feature por valor, uma vez que apenas existiam dois valores distintos para cada *feature* discreta. Este processo consistiu em criar um atributo por valor, uma vez que estas variáveis eram binárias.
- **Normalização** - Como os dados tinham diferentes domínios foi realizada uma normalização dos mesmos, utilizando a média e o desvio padrão.
- **Feature Selection / Reduction** - Como o dataset original tem um número elevado de *features* foram testadas técnicas de redução de *features*, de forma a obtermos um *dataset* menos complexo e a eliminar características pouco discriminativas. Foram testadas as duas técnicas apresentadas a seguir:
 - **Principal Component Analysis⁹ (PCA)** - Uma técnica de *feature reduction* é o PCA, que consiste na conversão dos atributos em novos atributos através de transformações ortogonais. Como esta técnica mudava o significado real das características não pode ser utilizada no âmbito deste projeto uma vez que pretendemos que a resposta contenha informação relativa à decisão realizada, para que o utilizador a possa analisar.
 - **Information Gain¹⁰ (IG)** - Uma vez que com a utilização do PCA não é possível preservar a informação original foi utilizado *Information Gain* para *feature selection* selecionando o top N de *features* mais relevantes. Foram sendo realizados testes com alguns valores diferentes para N de forma a se seleccionar o N que mais favorecia o sistema. Os valores de N utilizados variavam entre dezanove (19) e vinte e quatro (24). Esta técnica tem por base a entropia de um atributo para ordenar os atributos pela sua capacidade discriminativa.

⁹https://en.wikipedia.org/wiki/Principal_component_analysis

¹⁰https://en.wikipedia.org/wiki/Information_gain_ratio

Nas tabelas seguintes é possível observar os resultados dos testes realizados que permitiram selecionar a melhor configuração para o sistema. Para estes testes foram utilizados dois *datasets* criados durante a fase de *Data Sampling* (ponto 6.3.3). Os *datasets* selecionados foram os que tinham um rácio de 5/1 e 6/1, uma vez que estes *datasets* eram os que nos ofereciam uma maior percentagem de instâncias classificadas corretamente em ambas as classes e tinham os resultados mais equilibrados entre as classes.

Para a realização destes testes foi utilizado o classificador *Classification Tree* e foram sendo testadas diferentes configurações do sistema que incidiram principalmente sobre três aspetos:

- Variáveis Contínuas - Este primeiro ponto consistiu em testar o sistema com a transformação das variáveis discretas em contínuas, como foi descrito anteriormente. Nas tabelas 6.4 e 6.5 os resultados em que esta operação foi realizada estão identificados com a letra *C* na primeira posição.
- *Information Gain* - Para a seleção de *features* foi utilizada esta técnica e foram realizados vários testes com diferentes números de *features*. Os resultados identificados com o identificador contém o código *IG* e termina com um número que indica o número de *features* selecionadas.
- *Normalize* - Identifica os testes realizados com os dados normalizados, sendo que os resultados encontram-se identificados com a letra *N* antes do número.

Os identificadores de resultados que contém o carácter * são resultantes dos testes realizados com o *dataset* que contém um rácio de 6/1, enquanto que onde não aparece o asterisco são resultados obtidos com a utilização de um *dataset* com o rácio 5/1.

Tabela 6.4: Métricas de performance do sistema.

#	AUC	ACC	F1-Score	Precision	Recall	%Correct
Configuração Sistema	Variáveis Contínuas, <i>Information Gain</i>					
CIG19	75.7	76.5	86.2	98.4	76.6	76.5490
CIG20	76.2	76.3	86	98.5	76.3	76.2712
CIG21	76.1	76.7	86.3	98.5	76.8	76.7158
CIG22	75.4	76	85.8	98.4	76	75.9933
CIG23	74	75.9	85.7	98.2	76.1	75.8822
CIG24	73.7	76	85.8	98.2	76.2	75.9933
CIG19*	75	77.9	87.1	98.3	78.1	77.8550
CIG20*	76.3	77.1	86.6	98.5	77.2	77.1325
CIG21*	76.1	78.4	87.4	98.4	78.6	78.4107
CIG22*	76.6	78.4	87.5	98.5	78.7	78.4940
CIG23*	74.3	78.1	87.2	98.2	78.5	78.0772
CIG24*	73.1	77.6	86.9	98.1	78	77.5771
Configuração Sistema	<i>Information Gain</i>					
IG20	76.5	76.3	86	98.5	76.3	76.3268
IG21	76.6	76.5	86.1	98.5	76.5	76.4657
IG22	74.7	76.2	86	98.3	76.4	76.2434
IG20*	76.4	77.3	86.7	98.5	77.4	77.2992
IG21*	76.7	78.4	87.4	98.5	78.6	78.3829
IG22*	76.2	78.6	87.5	98.4	78.8	78.6052
Configuração Sistema	Variáveis Contínuas, <i>Information Gain, Normalize</i>					
CIGN20	76.9	76.5	86.2	98.6	76.5	76.5490
CIGN21	76.8	76.4	86.1	98.6	76.4	76.4101
CIGN20*	75.9	76.9	86.4	98.4	77	76.8825
CIGN21*	76.2	78	87.1	98.4	78.2	77.9939

Tabela 6.5: Instâncias identificadas corretamente

#	Diabetes	SemDiabetes	%Diabetes	%SemDiabetes	#Inst
Configuração Sistema	Variáveis Contínuas, <i>Information Gain</i>				
CIG19	125	2630	74.85	76.63	84558 5 / 1
CIG20	127	2618	76.05	76.28	
CIG21	126	2635	75.45	76.78	
CIG22	125	2610	74.85	76.05	
CIG23	120	2611	71.86	76.08	
CIG24	119	2616	71.26	76.22	
CIG19*	120	2682	71.86	78.15	98651 6 / 1
CIG20*	126	2650	75.45	77.21	
CIG21*	123	2699	73.65	78.64	
CIG22*	125	2700	74.85	78.67	
CIG23*	117	2693	70.06	78.47	
CIG24*	114	2678	68.26	78.03	
Configuração Sistema	<i>Information Gain</i>				
IG20	128	2619	76.65	76.31	84558 5 / 1
IG21	128	2624	76.65	76.46	
IG22	122	2622	73.05	76.4	
IG20*	126	2656	75.45	77.39	98651 6 / 1
IG21*	125	2696	74.85	78.55	
IG22*	123	2706	73.65	78.85	
Configuração Sistema	Variáveis Contínuas, <i>Information Gain, Normalize</i>				
CIGN20	129	2626	77.25	76.52	84558 5 / 1
CIGN21	129	2621	77.25	76.37	
CIGN20*	125	2642	74.85	76.98	98651 6 / 1
CIGN21*	124	2683	74.25	78.18	

Após a realização dos vários testes foi selecionado um *dataset* com um rácio de 5/1 (para cada instância da classe diabetes existem cinco instâncias da classe contrária). Também foi decidido que o sistema utilizará para *feature selection* o método *Information Gain* selecionando as 21 *top features*. Os resultados relativamente a esta configuração do sistema encontram-se selecionados com a cor verde nas tabelas 6.4 e 6.5. Quanto aos resultados assinalados com a cor amarela, são os melhores de cada uma das experiências realizadas com os diferentes parâmetros (*Feature Selection* e rácios). Existem algumas variantes de configuração do sistema que podem ter resultados semelhantes ou um pouco melhores que os resultados da configuração selecionada. Por sua vez essas configurações iriam necessitar da realização de mais

cálculos sobre os dados, como por exemplo o caso de se efetuar a normalização dos dados. Contudo podemos observar que quando comparamos os resultados da configuração selecionada com os resultados obtidos quando se aplicam essas transformações, resultado IG21 com o resultado CIGN20, podemos observar que a diferença no número de instâncias classificadas com a classe diabetes é de apenas uma instância e relativamente à classe contrária a diferença é de três instâncias.

6.3.5 Classificação

No sistema para a classificação foi selecionado o *widget classification tree*, que se encontra presente no software utilizado para o treino dos modelos. Este componente utiliza o algoritmo *Decision Tree Classifier* que se encontra presente na biblioteca *Sklearn*¹¹. O algoritmo *Decision tree* utiliza como critério de decisão a entropia da informação para selecionar a *feature* mais relevante para cada nó da árvore, conseguindo assim uma maior separação da informação.

Outros algoritmos ainda foram considerados para o problema, mas devido à restrição em que era desejável que os modelos fossem interpretáveis, isto é, a apresentação dos critérios que estão na origem de cada decisão não foi possível utilizar esses algoritmos. Os algoritmos que ainda estiveram em consideração para serem utilizados foram os seguintes:

- KNN - Este algoritmo utiliza os k vizinhos mais próximos para tomar uma decisão e classificar as instâncias tendo em conta as classes desses vizinhos.
- *Logistic Regression* - É um algoritmo estatístico que classifica uma instância tendo em conta a probabilidade dessa instância pertencer à classe em questão, baseado nas *features*.
- *Naive Bayesian Classifier* - Este algoritmo pertence à família dos classificadores probabilísticos, e utiliza a probabilidade de uma instância pertencer a um determinada classe tendo em conta as diferentes *features*, com a hipótese dessas *features* serem independentes.

Classification Tree

Para o treino do modelo de classificação foi construído um fluxo, com recurso ao software da *Orange*, que nos permitiu treinar, testar e afinar os diferentes parâmetros para selecionar a melhor configuração do sistema e assim obter os melhores resultados. A utilização deste software permite uma fácil alteração dos parâmetros e teste de vários classificadores, o que veio a ser uma mais valia para o projeto.

Na figura E.2, do Apêndice E, é possível observar o fluxo criado para o treino do modelo final. Neste fluxo são carregados os ficheiros de treino e teste do sistema e é realizado o treino do modelo que é guardado para ser utilizado posteriormente no sistema de classificação final. O fluxo é composto também por componentes que permitem analisar os resultados e os dados utilizados ao longo dos passos, assim como a árvore de decisão criada.

¹¹<http://scikit-learn.org/>

6.3.6 Conclusões

Após a afinação do modelo durante a fase de testes foi selecionada a melhor configuração para o sistema, sendo esta a seguinte:

- **Data Sampling** - O *dataset* de treino tem um rácio de 5/1
- **Feature Selection** - *Information Gain* selecionando as 21 *features* melhor classificadas.
- **Classificador** - *Classification Tree*, isto devido às necessidades de se obter informação sobre as *features* que levaram a uma determinada decisão.

Para a validação do sistema foi utilizado um novo *dataset* independente do *dataset* de treino e de teste e que é composto por 34769 instâncias. Nas tabelas 6.6 e 6.7 é possível observar os resultados da validação do sistema.

Tabela 6.6: Métricas de performance aquando da validação do modelo.

AUC	ACC	F1-Score	Precision	Recall	%Correct
84.4	92.8	96.1	98.5	93.8	92.7579

Tabela 6.7: Instâncias identificadas corretamente aquando da validação do modelo.

Diabetes	SemDiabetes	%Diabetes	%SemDiabetes	#Instances
1447	30804	75.0519	93.7974	34769

Com os resultados da validação podemos concluir que o facto destes serem superiores aos da fase de testes é relacionado com o tamanho e a origem dos *datasets*. Este melhoramento dos resultados é mais visível nos valores que são referentes à classe mais frequente, neste caso "Sem Diabetes".

Na figura 6.19 é possível observar os primeiros quatro níveis da árvore de decisão resultante do treino do modelo com a configuração selecionada. Na imagem podemos observar que os nós identificados com a cor vermelha são os nós pertencentes à classe "Sem Diabetes" e os nós identificados a azul são os nós referentes à classe "Diabetes". A cor varia conforme a percentagem de instâncias, pertencentes a cada uma das classes, que se encontram presentes em cada nó.

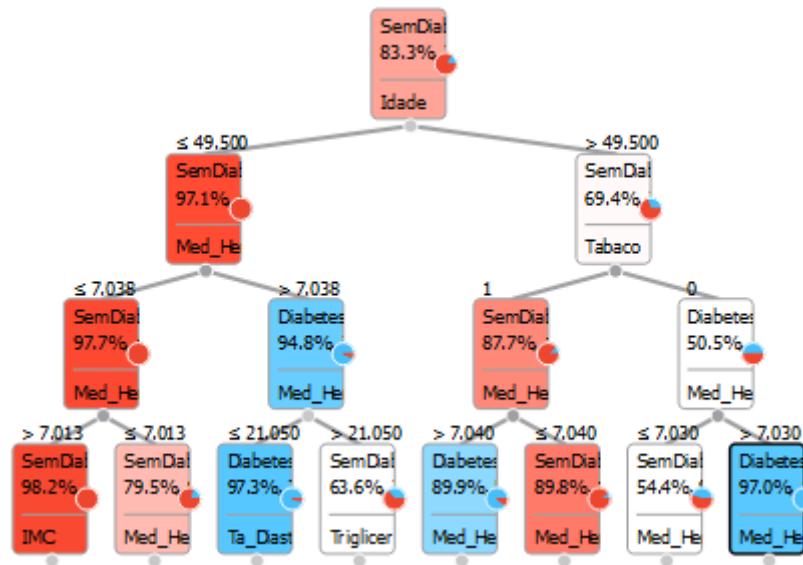


Figura 6.19: Primeiros quatro níveis da árvore de decisão.

Após a obtenção de um modelo de classificação com resultados minimamente satisfatórios foi guardado esse modelo e posteriormente criado o serviço de classificação como apresentado na arquitetura.

O serviço que vai responder aos pedidos para classificação de cada processo clínico foi construído recorrendo à tecnologia *Python*, pois assim foi possível utilizar as ferramentas presentes na biblioteca *Orange* e *Sklearn*.

O serviço criado recebe um pedido REST em que no corpo do pedido vai a instância a classificar, no formato JSON, conforme apresentado no Anexo F.1. A resposta ao pedido é devolvida também no formato de mensagem JSON, como é possível observar no exemplo do Anexo F.2.

CAPÍTULO 6. TRABALHO DESENVOLVIDO

Capítulo 7

Testes de Aceitação

Neste capítulo pretende-se apresentar o plano de testes seguido de forma a garantir que as funcionalidades implementadas estavam de acordo com o que era pedido. Para a validação do trabalho desenvolvido foram apenas realizados testes de aceitação a fim de validar se as funcionalidades referentes às *features* funcionam corretamente e se têm o comportamento esperado. Estes são testes manuais e foram sendo realizados à medida que cada *feature* era desenvolvida. Com a realização destes testes e a análise dos resultados, as *features* eram sendo fechadas após o resultado positivo nos mesmos.

Apenas foram realizados testes de aceitação, na medida em que estes testes nos indicam se uma determinada funcionalidade já se encontra completa ou não. Como o principal objetivo do projeto era validar uma ideia para evoluir, não se verificou a necessidade de serem realizados mais testes nesta fase inicial do projeto.

Na tabela 7.1 é possível consultar os testes realizados e os resultados obtidos.

Tabela 7.1: Conjunto de testes de aceitação realizados à linha do tempo.

#Testes	Descrição de Testes	UserStory	Resultado
1	Alternar entre os três momentos (Passado, Presente, Futuro).	Ge_US05	OK
2	Verificar se as fotografias dos utentes aparecem, caso existam, e caso contrário se aparecem as iniciais do nome.	Ge_US09	OK
3	Alternar entre elementos do agregado familiar e verificar se os dados obtidos correspondem ao elemento correto.	Ge_US09	OK
4	Testar iniciar aplicação e obter membros dos agregados familiares e os dados necessários.	Ge_US09	OK
5	Abrir a aplicação e verificar se a secção inicial é a secção referente ao momento do Presente.	Ge_US01	OK
6	Modificar as cores dos grupos de dados a apresentar na linha do tempo.	Ge_US07	OK

CAPÍTULO 7. TESTES DE ACEITAÇÃO

7	Alterar os grupos de eventos visíveis. Selecionar Patologias e MCDT apenas.	Ge_US08	OK
8	Verificar se os dados mostrados no passado são apenas dados de eventos que já ocorreram.	Pa_US01	OK
9	Visualizar o sumário clínico de um evento ao passar com o cursor do rato por cima.	Ge_US03	OK
10	Validar que as cores dos eventos são as cores definidas para o grupo ao qual o evento pertence.	Pa_US02, Pr_US05	OK
11	Ampliar/Reduzir o nível de zoom da linha do tempo.	Pa_US06, Fu_US04	OK
12	No momento do Presente verificar que a faixa central é apresentada com um fundo verde e as restantes com o fundo a branco.	Pr_US03	OK
13	Alterar o intervalo de tempo do momento do Presente e visualizar o resultado.	Pr_US06	OK
14	Navegar na linha do tempo através da utilização do <i>overview</i> .	Pa_US05	OK
15	Navegar para o evento anterior ao primeiro visível, e de seguida para o seguinte ao último visível.	Passado	OK
16	Tomar conhecimento do grupo a que um evento pertence visualizando a informação existente no sumário do evento.	Pa_US03	OK
17	Visualizar as atividades que o utente já tem agendadas para o futuro.	Fu_US01	OK
18	Visualizar uma atividade no momento do Futuro e verificar o seu estado e a sua data.	Fu_US03	OK
19	Marcar um evento e visualizar o evento marcado.	Ge_US06	Parcial, só marca o evento não é adicionado à lista dos eventos marcados.
20	Desmarcar um evento e verificar que esse evento foi desmarcado.	Ge_US06	OK
21	Visualizar uma atividade planeada para o futuro.	Fu_US05	US Não Implementada

CAPÍTULO 7. TESTES DE ACEITAÇÃO

22	Visualizar os dados clínicos dos últimos 3 meses.	Pr_US01	US Não Implementada
23	Visualizar eventos que se encontrem ativos no momento atual.	Pr_US07	US Não Implementada
24	Abrir janela com detalhes de um determinado evento clínico.	Ge_US04	US Não Implementada
25	Processar uma atividade planeada.	Fu_US02	US Não Implementada
26	Visualizar se os valores do processo clínico estão bem classificados (Acima do Normal, Normal e Abaixo do Normal).	Pr_US02	US Não Implementada

CAPÍTULO 7. TESTES DE ACEITAÇÃO

Capítulo 8

Conclusões Finais e Trabalho Futuro

Este capítulo marca o fim deste projeto de estágio na *MedicineOne*, e é um bom momento para fazer um apanhado destes últimos nove meses em que estive envolvido neste projeto. Com isto pretende-se analisar o trabalho desenvolvido e o que fica como trabalho futuro.

Nas duas secções seguintes é apresentado em primeiro lugar o trabalho desenvolvido e são mostrados alguns aspetos que foram surgindo ao longo do estágio. Na última secção é apresentada uma linha de seguimento deste projeto identificando os pontos mais relevantes, do projeto, que ficam para serem realizados na continuação do mesmo.

8.1 Conclusões finais

O principal objetivo deste projeto era analisar a possibilidade de mostrar o processo clínico, de um utente, numa nova forma de visualização diferente de como atualmente este é visto. Com isto, era pretendido que fosse implementada a visualização dos dados clínicos com recurso a uma linha do tempo.

Devido à possível simplicidade deste componente, o estágio foi dividido em duas partes, sendo que a primeira parte consistia na implementação da linha do tempo para a visualização do processo clínico e a segunda parte consistiu no estudo de um mecanismo que permitisse enriquecer a informação que o utilizador tem disponível na linha do tempo.

No caso da linha do tempo, esta encontra-se dividida em três momentos, como mostrado ao longo deste relatório, e em cada dos momentos é apresentada a informação referente ao mesmo. Sendo que os requisitos, relativos a cada um dos momentos, eram semelhantes e não acrescentavam complexidade ao projeto, com o avançar do semestre foi tomada a decisão de parar a implementação da linha do tempo e realizar a segunda parte do estágio.

Relativamente à segunda parte do estágio, esta consistiu num estudo do problema analisando a informação que se encontrava disponível para utilização nesta segunda parte. Após a análise dessa informação foi necessária a criação de *datasets* para serem utilizados pelo sistema. Com isto foi possível criar um modelo que nos indica se um determinado utente poderá vir a contrair *diabetes tipo II* e quais os valores clínicos que podem estar na sua origem.

No decorrer deste projeto foram surgindo algumas dificuldades que se deveram a vários aspetos relacionados diretamente e indiretamente com o estágio. A principal dificuldade foi a utilização de novas tecnologias com as quais o estagiário teve de trabalhar que poderão ter levado a pequenos atrasos na execução do projeto que no final levaram a que nem todos os requisitos relativos à primeira fase fossem terminados.

Em forma de concluir este ponto, podemos dizer que se conseguiram atingir os objetivos uma vez foi possível mostrar visivelmente os dados clínicos numa linha do tempo como era pretendido e ainda conseguimos enriquecer a informação presente na mesma apresentando visualmente se o paciente em análise pode vir a contrair a patologia, ou não.

8.2 Trabalho futuro

Por impossibilidade da realização completa de todos os objetivos pretendidos para o trabalho, na duração do período de estágio, deixam-se aqui algumas recomendações para trabalho futuro na continuação deste projeto.

Em primeiro lugar, deixa-se aqui a sugestão de ao finalizar o desenvolvimento da linha do tempo, que também sejam adicionados novos tipos de dados à linha do tempo permitindo assim uma mais completa visualização do processo clínico. Após a finalização da linha do tempo é pretendido que esta seja integrada no sistema *MedicineOne 8*.

Também se encontra planeada a implementação do requisito *Pa_US04*, que consiste em identificar relações do tipo causa-efeito nos dados que já se encontram presentes no processo clínico, por exemplo caso um paciente já tenha contraído uma determinada patologia e esta seja relacionada com a interação de dois eventos ocorridos anteriormente, a relação deverá ser identificada graficamente à semelhança de como são identificadas as causas de um utente poder ou não contrair diabetes (referente à segunda parte deste estágio).

Relativamente à segunda parte deste estágio fica a sugestão de melhorar o modelo de classificação de forma a que se obtenha um grau de confiança maior. Também fica a sugestão de estender esta segunda parte a outras patologias que não apenas a *diabetes tipo II* permitindo assim um maior enriquecimento dos dados que o utilizador tem ao seu dispor.

Bibliografia

- [1] L. Chittaro, “Information visualization and its application to medicine,” *Artificial intelligence in medicine*, vol. 22, no. 2, pp. 81–88, 2001.
- [2] “Information visualization.” https://en.wikipedia.org/wiki/Information_visualization. Acesso: 2016-01-14.
- [3] K. K. Gandhi and N. B. Prajapati, “Study of diabetes prediction using feature selection and classification,” *International Journal of Engineering Research and Technology*, vol. 3, no. 2, 2014.
- [4] K. Plis, R. Bunescu, C. Marling, J. Shubrook, and F. Schwartz, “A machine learning approach to predicting blood glucose levels for diabetes management,” *Modern Artificial Intelligence for Health Analytics. Papers from the AAAI-14*, 2014.
- [5] J. A. Cruz and D. S. Wishart, “Applications of machine learning in cancer prediction and prognosis,” *Cancer informatics*, vol. 2, 2006.
- [6] I. Mandal and N. Sairam, “New machine-learning algorithms for prediction of parkinson’s disease,” *International Journal of Systems Science*, vol. 45, no. 3, pp. 647–666, 2014.
- [7] F. S. Roque, L. Slaughter, and A. Tkatšenko, “A comparison of several key information visualization systems for secondary use of electronic health record content,” in *Proceedings of the NAACL HLT 2010 Second Louhi Workshop on Text and Data Mining of Health Documents*, pp. 76–83, Association for Computational Linguistics, 2010.
- [8] C. Plaisant, B. Milash, A. Rose, S. Widoff, and B. Shneiderman, “Lifelines: visualizing personal histories,” in *Proceedings of the SIGCHI conference on Human factors in computing systems*, pp. 221–227, ACM, 1996.
- [9] C. Plaisant, R. Mushlin, A. Snyder, J. Li, D. Heller, B. Shneiderman, *et al.*, “Lifelines: Using visualization to enhance navigation and analysis of patient records,” in *In Proceedings of the 1998 American Medical Informatic Association Annual Fall Symposium*, 1998.
- [10] C. Plaisant, D. Heller, J. Li, B. Shneiderman, R. Mushlin, and J. Karat, “Visualizing medical records with lifelines,” *CHI 98 Cconference Summary on Human Factors in Computing Systems*, 1998.
- [11] R. Bradley and S. Nestler, “Adapting lifelines to army officer personnel processes,”

- [12] C. Hallett, “Multi-modal presentation of medical histories,” in *Proceedings of the 13th international conference on Intelligent user interfaces*, pp. 80–89, ACM, 2008.
- [13] S. B. Martins, Y. Shahar, M. Galperin, H. Kaizer, D. Goren-Bar, D. McNaughton, L. V. Basso, and M. K. Goldstein, “Evaluation of knave-ii: a tool for intelligent query and exploration of patient data,” *Medinfo*, vol. 11, no. Pt 1, pp. 648–52, 2004.
- [14] Y. Shahar, D. Goren-Bar, M. Galperin, D. Boaz, and G. Tahan, “Knave-ii: A distributed architecture for interactive visualization and intelligent exploration of time-oriented clinical data,” *Proceedings of Intelligent Data Analysis in Medicine and Pharmacology, Protaras, Cyprus*, 2003.
- [15] D. Goren-Bar, Y. Shahar, M. Galperin-Aizenberg, D. Boaz, and G. Tahan, “Knave ii: the definition and implementation of an intelligent tool for visualization and exploration of time-oriented clinical data,” in *Proceedings of the working conference on Advanced visual interfaces*, pp. 171–174, ACM, 2004.
- [16] A. A. Bui, D. R. Aberle, and H. Kangarloo, “Timeline: visualizing integrated patient records,” *Information Technology in Biomedicine, IEEE Transactions on*, vol. 11, no. 4, pp. 462–473, 2007.
- [17] S. Miksch, R. Kosara, Y. Shahar, and P. Johnson, “Asbruvew: Visualization of time-oriented, skeletal plans,” 1998.
- [18] A. Seyfang and S. Miksch, *Asgaard’s contribution to the guideline representation comparison*. na, 2002.
- [19] “Systmone website.” <http://www.tpp-uk.com/products/systmone/>. Acesso: 2015-10-25.
- [20] “Manual de utilizador do *SystemOne Patient Timeline*.” <http://www.bradfordvts.co.uk/wp-content/onlineresources/0402imt/systm%20one%20patient%20timeline.pdf>. Acesso: 2015-10-25.
- [21] “Feature driven development and agile modeling.” <http://www.agilemodeling.com/essays/fdd.htm>. Acesso: 2015-11-16.
- [22] E. Nickell, “Feature driven development.” <http://www.slideserve.com/elon/feature-driven-development>. Acesso: 2015-11-16.
- [23] “Feature-driven development.” https://en.wikipedia.org/wiki/Feature-driven_development. Acesso: 2015-11-16.
- [24] M. Vieira, *Slides das aulas de Gestão de Projetos*. Universidade de Coimbra, Coimbra, Portugal, Setembro-Dezembro 2014.
- [25] A. J. Dorofee, J. A. Walker, C. J. Alberts, R. P. Higuera, and R. L. Murphy, *Continuous Risk Management Guidebook*, ch. Appendix A-18, pp. 391–398. Carnegie Mellon University, 1996.

- [26] “Writing user stories.” <https://www.gov.uk/service-manual/agile/writing-user-stories.html>. Acesso: 2015-11-30.
- [27] “The easy way to writing good user stories.” <http://codesqueeze.com/the-easy-way-to-writing-good-user-stories/>. Acesso: 2015-11-30.
- [28] “Your best agile user story.” <http://www.alexandercowan.com/best-agile-user-story/>. Acesso: 2015-11-30.
- [29] “User story template advantages.” <http://www.mountaingoatsoftware.com/blog/advantages-of-the-as-a-user-i-want-user-story-template>. Acesso: 2015-11-30.
- [30] “Diário da república, lei nº 12/2005.” <https://www.cnpd.pt/bin/legis/nacional/Lei12-2005.pdf>. Acesso: 2015-11-25.
- [31] B. Drohan, G. Grinstein, and K. Hughes, “Oncology lifeline—a timeline tool for the interdisciplinary management of breast cancer patients in a surgical clinic,” in *Information Visualisation (IV), 2010 14th International Conference*, pp. 233–238, IEEE, 2010.
- [32] B. Harrison, R. Owen, and R. Baecker, “Timelines an interactive system for the collection and visualization of temporal data,” 1994.
- [33] P. H. Nguyen, K. Xu, R. Walker, and B. W. Wong, “Timesets: Timeline visualization for sensemaking,”
- [34] L. Chittaro, “Visualization of patient data at different temporal granularities on mobile devices,” 2006.
- [35] M. Jensen, “Visualizing complex semantic timelines,” 2003.
- [36] O. Alonso, R. Baeza-Yates, and M. Gertz, “Exploratory search using timelines,” in *SIGCHI 2007 Workshop on Exploratory Search and HCI Workshop*, no. 1, pp. 1–4, 2007.
- [37] K. Kurihara, D. Vronay, and T. Igarashi, “Flexible timeline user interface using constraints,” *CHI’05 Extended Abstracts on Human Factors in Computing Systems*, 2005.
- [38] R. B. Allen, “A focus-context browser for multiple timelines,” 2005.
- [39] V. Kumar, R. Furuta, and R. B. Allen, “Metadata visualization for digital libraries: interactive timeline editing and review,” 1998.
- [40] M. Burch, F. Beck, and S. Diehl, “Timeline trees: Visualizing sequences of transactions in information hierarchies,” 2008.
- [41] P. André, M. L. Wilson, A. Russell, D. A. Smith, A. Owens, *et al.*, “Continuum: designing timelines for hierarchies, relationships and scale,” 2007.
- [42] R. Dachsel and M. Weiland, “Timezoom: a flexible detail and context timeline,” in *CHI’06 Extended Abstracts on Human Factors in Computing Systems*, pp. 682–687, ACM, 2006.

- [43] M. MacEachern, W. Townsend, K. Young, and G. Rana, “Librarian integration in a four-year medical school curriculum: a timeline,” *Medical reference services quarterly*, vol. 31, no. 1, pp. 105–114, 2012.
- [44] C. Stab, K. Nazemi, and D. W. Fellner, “Sematime-timeline visualization of time-dependent relations and semantics,” in *Advances in Visual Computing*, pp. 514–523, Springer, 2010.
- [45] M. Arguello, M. Fernandez-Prieto, and J. Des, “Extracting and visualising clinical statements from electronic health records,” in *Research and Development in Intelligent Systems XXX*, pp. 307–320, Springer, 2013.
- [46] A. A. Bui and R. K. Taira, *Medical imaging informatics*. Springer Science & Business Media, 2009.
- [47] C. Combi, E. Keravnou-Papailiou, and Y. Shahar, *Temporal information systems in medicine*. Springer Science & Business Media, 2010.
- [48] A. Rind, T. D. Wang, A. Wolfgang, S. Miksch, K. Wongsuphasawat, C. Plaisant, and B. Shneiderman, “Interactive information visualization to explore and query electronic health records,” *Foundations and Trends in Human-Computer Interaction*, vol. 5, no. 3, pp. 207–298, 2011.
- [49] B. Cabral, *Slides das aulas de Arquitetura de Software*. Universidade de Coimbra, Coimbra, Portugal, Fevereiro-Junho 2015.
- [50] J. M. De Sa, *Pattern recognition: concepts, methods and applications*. Springer Science & Business Media, 2012.

Apêndice A

Mockups Iniciais

A.1 Presente

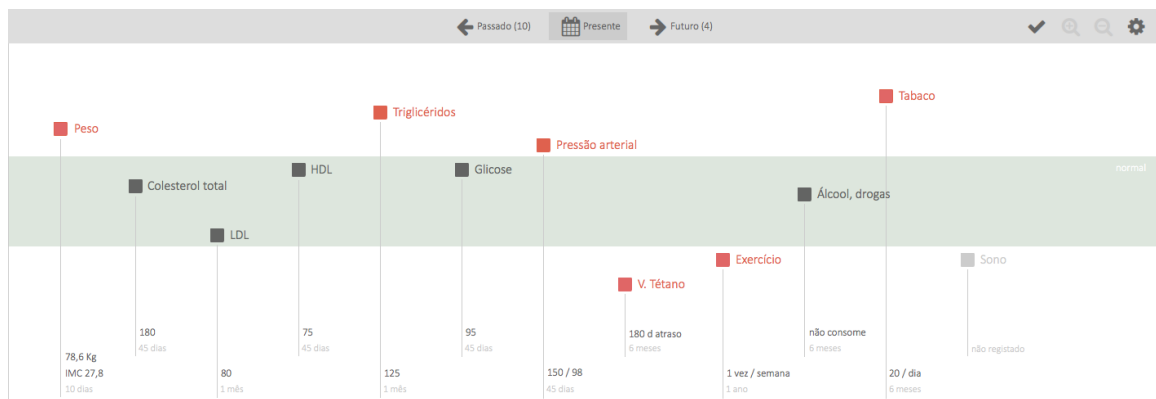


Figura A.1: Mockup inicial para o presente

A.2 Futuro

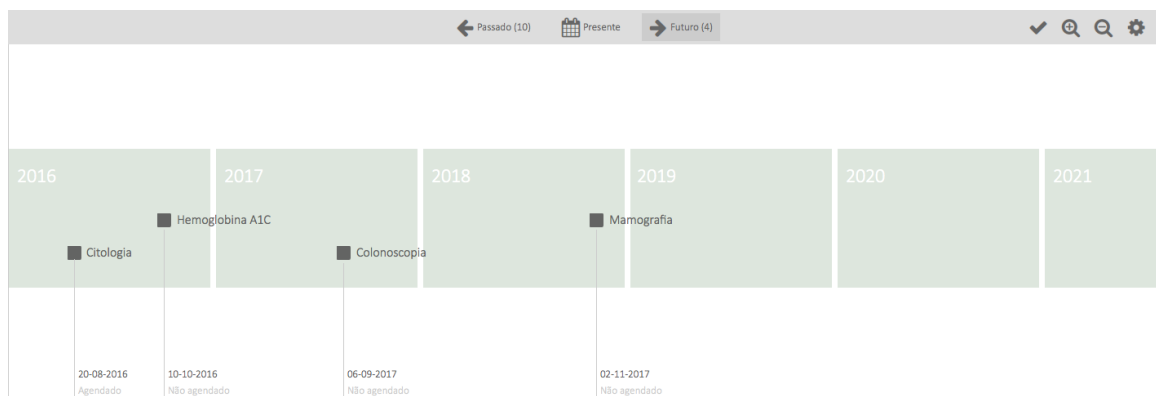


Figura A.2: Mockup inicial para o futuro

A.3 Passado

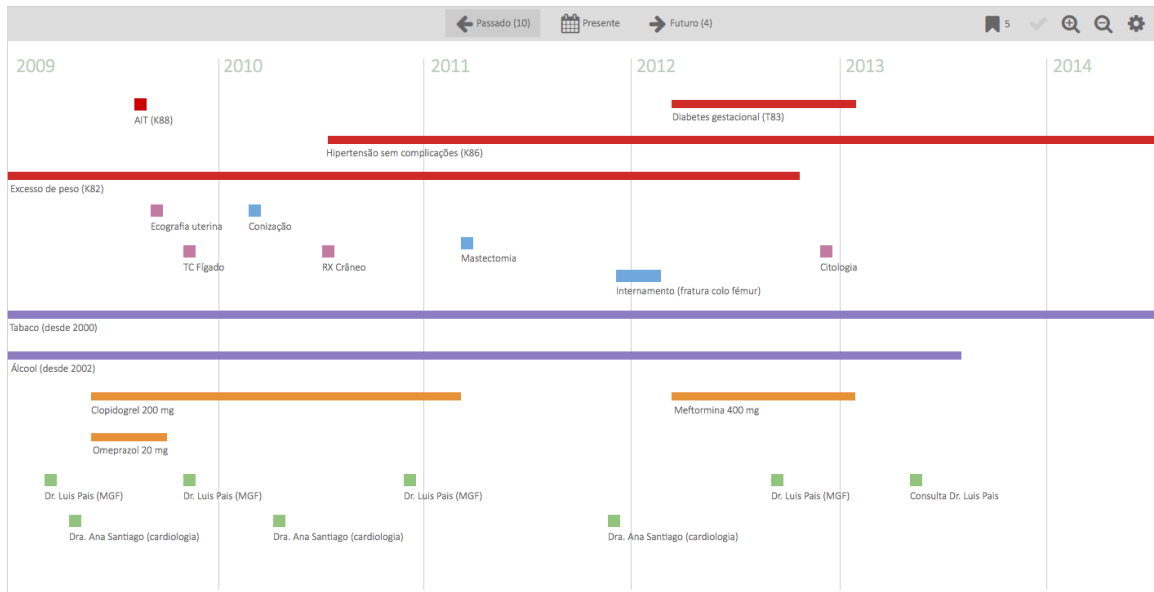


Figura A.3: *Mockup* inicial para o passado

Apêndice B

Requisitos Funcionais

B.1 Gerais ao Sistema

ID: Ge_US01	Prioridade: <i>Must</i>	Estado: Completo
Dependências: N/A		
Título: Apresentar a secção do Presente na abertura do módulo		
Enquanto Profissional clínico		
Quero visualizar a secção do presente ao entrar no módulo		
Para efetuar a análise do estado de saúde do utente		

ID: Ge_US02	Prioridade: <i>Nice</i>	Estado: Removido
Dependências: Pr_US04; Ge_US09		
Título: Visualizar as notificações dos elementos do agregado familiar		
Enquanto Profissional clínico		
Quero visualizar as notificações dos elementos do agregado familiar		
Para efetuar o seu processamento		
Descrição: Os elementos que tenham notificações deverão aparecer com um marcador no botão de acesso.		

APÊNDICE B. REQUISITOS FUNCIONAIS

ID: Ge_US03	Prioridade: <i>Must</i>	Estado: Completo
Dependências: Pa_US01		
Título: Mostrar sumário de um evento ao passar com o cursor do rato (<i>Tooltip</i>)		
Enquanto Profissional clínico		
Quero poder consultar mais informação ao passar com o rato por cima de um evento		
Para ver o seu sumário clínico		
Descrição: O sumário deverá aparecer numa <i>tooltip</i> e deverá conter os valores, datas e outra informação que possa ser considerada relevante para o evento.		

ID: Ge_US04	Prioridade: <i>Nice</i>	Estado: Não Implementado
Dependências: Pr_US01; Pa_US01; Fu_US01		
Título: Ao fazer duplo-clique sobre um evento mostrar, numa janela, a informação detalhada		
Enquanto Profissional clínico		
Quero visualizar mais informação relativa a um evento, ao fazer duplo-clique no mesmo		
Para ver toda a informação detalhada desse evento		
Descrição: A informação será mostrada numa nova janela.		

ID: Ge_US05	Prioridade: <i>Must</i>	Estado: Completo
Dependências: N/A		
Título: Alternar entre os diferentes momentos		
Enquanto Profissional clínico		
Quero alternar entre os diversos momentos do processo clínico do utente		
Para analisar o histórico e/ou o futuro do processo clínico do utente		

APÊNDICE B. REQUISITOS FUNCIONAIS

ID: Ge_US06	Prioridade: <i>Nice</i>	Estado: Parcial
Dependências: Ge_US05		
Título: Introduzir marcadores para identificar ocorrências relevantes		
Enquanto Profissional clínico Quero marcar ocorrências relevantes Para referência futura		
Descrição: Os eventos marcados devem aparecer assinalados na linha do tempo e na lista dos eventos marcados.		

ID: Ge_US07	Prioridade: <i>Must</i>	Estado: Completo
Dependências: N/A		
Título: Modificar o código de cores		
Enquanto Profissional clínico Quero alterar o código de cores utilizado Para facilitar a associação de cada evento ao respectivo grupo		

ID: Ge_US08	Prioridade: <i>Must</i>	Estado: Completo
Dependências: Ge_US07		
Título: Selecionar grupos de eventos visíveis		
Enquanto Profissional clínico Quero alterar os grupos de eventos visíveis Para simplificar a visualização dos dados importantes para mim		

ID: Ge_US09	Prioridade: <i>Must</i>	Estado: Completo
Dependências: N/A		
Título: Alternar entre os elementos do agregado familiar e apresentar a respetiva informação		
Enquanto Profissional clínico Quero visualizar outro elemento do agregado familiar Para analisar o seu estado de saúde		

B.2 Presente

ID: Pr_US01	Prioridade: <i>Nice</i>	Estado: Não Implementado
Dependências: Pr_US03; Pr_US06		
Título: Apresentar, inicialmente, a informação mais atual de cada setor do processo clínico do utente dos últimos 3 meses		
Enquanto Profissional clínico		
Quero visualizar os eventos mais recentes do processo clínico do utente		
Para analisar o estado de saúde do utente		
Descrição: Apenas está visível o último registo de cada item, que ocorreu no intervalo de tempo definido.		

ID: Pr_US02	Prioridade: <i>Nice</i>	Estado: Não Implementado
Dependências: Pr_US01		
Título: Apresentar os valores em faixas separadas segundo a sua classificação (Acima do normal, normal e abaixo do normal)		
Enquanto Profissional clínico		
Quero visualizar os eventos apresentados separados em três categorias conforme o seu estado		
Para uma melhor e rápida análise dos mesmos		
Descrição: Os valores devem estar separados nas seguintes categorias: Acima do Normal, Normal e Abaixo do Normal.		

ID: Pr_US03	Prioridade: <i>Must</i>	Estado: Completo
Dependências: N/A		
Título: As 3 faixas onde são representados os valores devem ter as seguintes cores, faixa central a verde, a faixa superior e a inferior devem ter um fundo branco		
Enquanto Profissional clínico		
Quero uma distinção entre os valores normais e anormais		
Para uma rápida análise dos valores representados na linha do tempo		
Descrição: A faixa dos valores normais deve ter um fundo verde, enquanto as outras faixas têm o fundo branco.		

APÊNDICE B. REQUISITOS FUNCIONAIS

ID: Pr_US04	Prioridade: <i>Must</i>	Estado: Removido
Dependências: Pr_US01		
Título: Mostrar alertas do utente e permitir o seu processamento		
Enquanto Profissional clínico Quero visualizar os alertas existentes no processo clínico Para realizar o seu processamento		

ID: Pr_US05	Prioridade: <i>Nice</i>	Estado: Completo
Dependências: Ge_US07		
Título: Apresentar dados com recurso a código de cores, pelo tipo de informação		
Enquanto Profissional clínico Quero visualizar os eventos, assinalados com o código de cores definido Para um fácil reconhecimento do grupo/categoria ao qual pertencem		

ID: Pr_US06	Prioridade: <i>Must</i>	Estado: Completo
Dependências: Pr_US03; Ge_US07		
Título: Personalizar o intervalo de tempo dos dados a visualizar		
Enquanto Profissional clínico Quero alterar o intervalo de tempo visível Para realizar uma melhor análise do processo clínico		

ID: Pr_US07	Prioridade: <i>Nice</i>	Estado: Não Implementado
Dependências: Pr_US01		
Título: Listar eventos que se encontrem ativos no momento		
Enquanto Profissional clínico Quero visualizar uma lista com os eventos por terminar Para conhecer os eventos que estão a ocorrer		

B.3 Passado

ID: Pa_US01	Prioridade: <i>Must</i>	Estado: Completo
Dependências: Ge_US05		
Título: Apresentar o histórico clínico do utente		
Enquanto Profissional clínico		
Quero visualizar o histórico clínico do utente		
Para analisar as ocorrências mais antigas		

ID: Pa_US02	Prioridade: <i>Must</i>	Estado: Completo
Dependências: Pa_US01; Ge_US07		
Título: Apresentar dados com recurso a código de cores, pelo tipo de informação		
Enquanto Profissional clínico		
Quero visualizar os eventos, assinalados com o código de cores definido		
Para um fácil reconhecimento do grupo/categoria ao qual pertencem		

ID: Pa_US03	Prioridade: <i>Nice</i>	Estado: Completo
Dependências: Pa_US02		
Título: Mostrar legenda do código de cores utilizado		
Enquanto Profissional clínico		
Quero visualizar a legenda das cores utilizadas		
Para saber qual o grupo que cada cor representa		

ID: Pa_US04	Prioridade: <i>Wishful</i>	Estado: Parcial
Dependências: Pa_US01		
Título: Assinalar graficamente relacionamentos do tipo causa-efeito entre eventos		
Enquanto Profissional clínico		
Quero visualizar as relações, do tipo causa-efeito, entre eventos		
Para perceber quais os eventos que podem ter estado na origem de um outro evento		

APÊNDICE B. REQUISITOS FUNCIONAIS

ID: Pa_US05	Prioridade: <i>Nice</i>	Estado: Completo
Dependências: Pa_US01		
Título: Deverá estar visível um <i>overview</i> de todos os eventos da linha do tempo		
<p>Enquanto Profissional clínico</p> <p>Quero visualizar um <i>overview</i> da linha do tempo</p> <p>Para ter conhecimento de até onde os eventos se estendem no histórico do utente</p>		

ID: Pa_US06	Prioridade: <i>Must</i>	Estado: Completo
Dependências: Pa_US01		
Título: Permitir realizar operações de zoom-in e zoom-out		
<p>Enquanto Profissional clínico</p> <p>Quero visualizar de forma mais detalhada, ou geral os eventos da linha do tempo</p> <p>Para uma melhor análise dos eventos ocorridos</p>		
Descrição: Deve ser possível mudar as escalas temporais com recurso a botões de ampliar e reduzir ou com a combinação “ <i>Crtl + MouseWheel</i> ”.		

B.4 Futuro

ID: Fu_US01	Prioridade: <i>Nice</i>	Estado: Completo
Dependências: Ge_US05		
Título: Apresentar atividades agendadas para o futuro por ordem cronológica		
Enquanto Profissional clínico		
Quero visualizar as atividades marcadas por ordem cronológica		
Para saber quais são as próximas atividades a realizar pelo utente		

ID: Fu_US02	Prioridade: <i>Nice</i>	Estado: Não Implementado
Dependências: Fu_US01		
Título: Processar atividades futuras		
Enquanto Profissional clínico		
Quero efetuar o processamento das próximas atividades futuras		
Para efetuar as respetivas requisições da mesma		
Descrição: O processamento pode ser realizado com recurso ao clique com o botão direito do rato ou no botão de processar existente.		

ID: Fu_US03	Prioridade: <i>Nice</i>	Estado: Completo
Dependências: Fu_US01		
Título: Identificar as atividades com informação relativa ao seu estado (Agendada ou Planeada) e data		
Enquanto Profissional clínico		
Quero visualizar o estado e data das atividades futuras		
Para saber quando as atividades serão realizadas e o seu estado (Agendada ou planeada)		
Descrição: As atividades futuras que se encontram agendadas devem aparecer com uma cor e as que se encontram planeadas com outra (Tons de cinzento).		

APÊNDICE B. REQUISITOS FUNCIONAIS

ID: Fu_US04	Prioridade: <i>Nice</i>	Estado: Completo
Dependências: N/A		
Título: Permitir realizar operações de zoom-in e zoom-out		
Enquanto Profissional clínico		
Quero visualizar de forma mais detalhada, ou geral os eventos da linha do tempo		
Para uma melhor análise dos eventos ocorridos		
Descrição: Deve ser possível mudar as escalas temporais com recurso a botões de ampliar e reduzir ou com a combinação “ <i>Ctrl + MouseWheel</i> ”.		

ID: Fu_US05	Prioridade: <i>Nice</i>	Estado: Não Implementado
Dependências: Ge_US05		
Título: Apresentar atividades que se encontram planeadas para o futuro		
Enquanto Profissional clínico		
Quero visualizar as atividades planeadas por ordem cronológica		
Para efetuar o agendamento da sua realização		
Descrição: Atividades planeadas são as atividades que ainda não se encontram marcadas, como por exemplo exames de rotina, entre outras.		

APÊNDICE B. REQUISITOS FUNCIONAIS

Apêndice C

Ordem de Implementação do Requisitos

APÊNDICE C. ORDEM DE IMPLEMENTAÇÃO DO REQUISITOS

Tabela C.1: Listagem ordenada dos requisitos pela ordem a implementar

#	ID	Prioridade	Dependências
01	Ge_US05	Must	N/A
02	Ge_US09	Must	N/A
03	Ge_US01	Must	N/A
04	Ge_US07	Must	N/A
05	Ge_US08	Must	Ge_US07
06	Pa_US01	Must	Ge_US05
07	Ge_US03	Must	Pa_US01
08	Pa_US02	Must	Pa_US01; Ge_US07
09	Pa_US06	Must	Pa_US01
10	Pr_US03	Must	N/A
11	Pr_US06	Must	Pr_US03; Ge_US07
12	Pa_US05	Nice	Pa_US01
13	Pa_US03	Nice	Pa_US02
14	Fu_US04	Nice	N/A
15	Fu_US01	Nice	Ge_US05
16	Fu_US03	Nice	Fu_US01
17	Pr_US05	Nice	Ge_US07
18	Ge_US06	Nice	Ge_US05
19	Fu_US05	Nice	Ge_US05
20	Pr_US01	Nice	Pr_US03; Pr_US06
21	Pr_US07	Nice	Pr_US01
22	Ge_US04	Nice	Pr_US01; Pa_US01; Fu_US01
23	Fu_US02	Nice	Fu_US01
24	Pr_US02	Nice	Pr_US01
25	Pa_US04	Wishful	Pa_US01
26	Pr_US04 ¹	Must	Pr_US01
27	Ge_US02 ¹	Must	Pr_US04; Ge_US09

¹Requisito excluído do projeto, por mudanças na estrutura de alertas da aplicação de destino.

Apêndice D

Estimativas

APÊNDICE D. ESTIMATIVAS

Tabela D.1: Estimativas realizadas para cada requisito (valores em horas)

ID	Best Case	Most Likely	Worst Case	Expected
Ge_US01	3	5	10	5.5
Ge_US02	8	12	20	12.6667
Ge_US03	4	6	12	6.6667
Ge_US04	16	20	32	21.3333
Ge_US05	5	8	12	8.1667
Ge_US06	8	12	20	12.6667
Ge_US07	10	14	22	14.6667
Ge_US08	6	10	16	10.3333
Ge_US09	16	24	36	24.6667
Pr_US01	24	32	48	33.3333
Pr_US02	8	12	20	12.6667
Pr_US03	4	6	10	6.3333
Pr_US04	6	10	20	11
Pr_US05	8	12	20	12.6667
Pr_US06	6	8	12	8.3333
Pr_US07	12	16	24	16.6667
Pa_US01	32	50	80	52
Pa_US02	8	12	20	12.6667
Pa_US03	6	8	12	8.3333
Pa_US04	56	80	110	81
Pa_US05	20	24	32	24.6667
Pa_US06	6	10	16	10.3333
Fu_US01	16	20	28	20.6667
Fu_US02	16	20	28	20.6667
Fu_US03	8	12	20	12.6667
Fu_US04	6	10	16	10.3333
Fu_US05	16	20	28	20.6667
	334	473	724	495.6667

Apêndice E

Fluxo de Treino do Modelo

Para a criação do fluxo apresentado, na figura E.2, foi utilizado o *software Orange Canvas*¹.

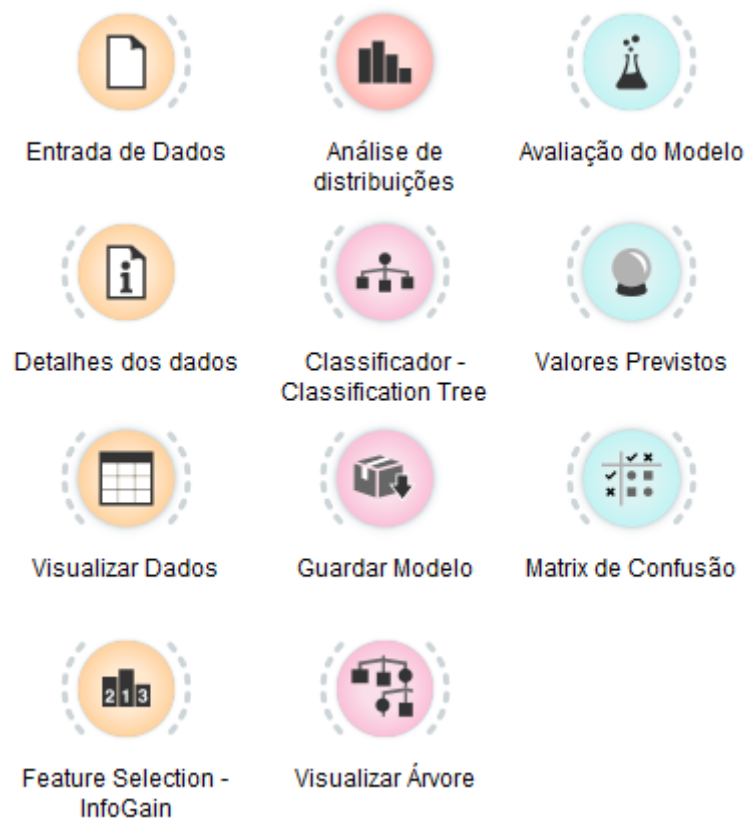


Figura E.1: Legenda dos componentes utilizados no fluxo da figura E.2.

¹<http://orange.biolab.si/>

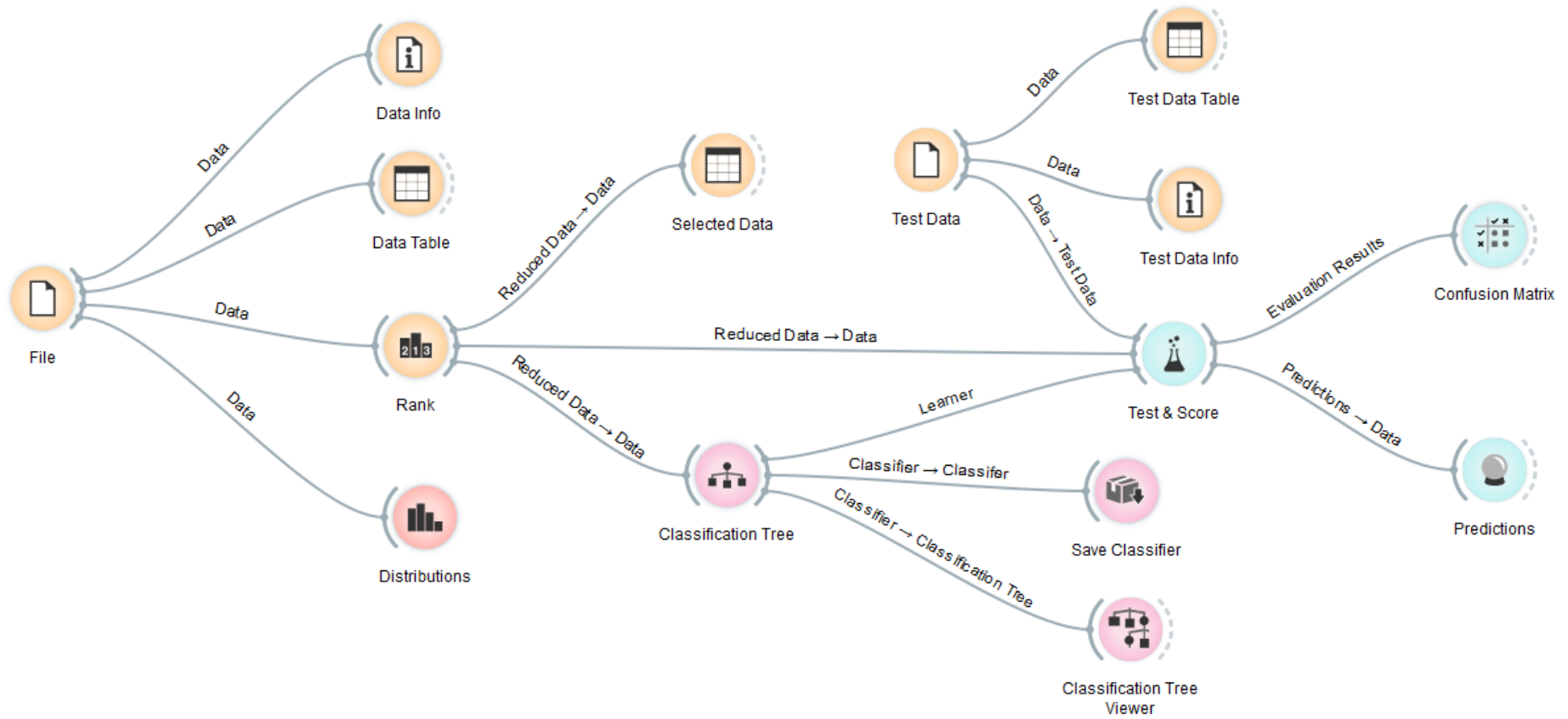


Figura E.2: Fluxo criado para o treino do modelo.

Apêndice F

Classificador Webservice API

O serviço que identifica se um paciente pode ou não contrair diabetes é um serviço REST em que recebe os dados por JSON e retorna a resposta também em JSON com a decisão e as regras que levaram à mesma. Nos pontos seguintes é possível observar a estrutura de um pedido e a estrutura da resposta ao mesmo.

F.1 Pedido

```
{
  "instance": "1,0,0,0,0,0,0,0,84,159,74,29.271,104,122.571
    4,44,67,89,97.1111,45,47.1111,130,110,168,161.5556,7.
    1,6.875,91,95.8571,1.1,0.97,13.9,13.75"
}
```

F.2 Resposta

```
{
  "classification": "Diabetes",
  "likelihood": {
    "diabetic": 0.8,
    "nondiabetic": 0.2
  },
  "decisionRules": [
    {
      "param": "Idade",
      "value": 49.5,
      "condition": ">",
      "ruleId": "0",
      "node": "0"
    },
    {
      "param": "Tabaco",
      "value": 1,
      "condition": "=",

```

```

"ruleId": "5",
"node": "1298"
},
{
"param": "Med_Hemoglobina_A1c",
"value": 7.0401999950408936,
"condition": "<=",
"ruleId": "12",
"node": "1299"
},
{
"param": "Med_Hemoglobina_A1c",
"value": 7.013122797012329,
"condition": "<=",
"ruleId": "12",
"node": "1300"
},
{
"param": "Med_Hemoglobina_A1c",
"value": 6.125849962234497,
"condition": ">",
"ruleId": "12",
"node": "1301"
},
{
"param": "Med_Hemoglobina_A1c",
"value": 6.529150009155273,
"condition": ">",
"ruleId": "12",
"node": "1669"
},
{
"param": "Med_Glucose",
"value": 114.42499923706055,
"condition": "<=",
"ruleId": "1",
"node": "1807"
},
{
"param": "Med-Ta_Diastolica",
"value": 78.88195037841797,
"condition": "<=",
"ruleId": "15",
"node": "1808"
},
{
"param": "Med-Ta_Sistolica",
"value": 144.5,

```

```
"condition": "<=",
"ruleId": "3",
"node": "1809"
},
{
  "param": "Med_Colesterol_Total",
  "value": 180.81739807128906,
  "condition": "<=",
  "ruleId": "23",
  "node": "1810"
},
{
  "param": "Triglicerides",
  "value": 92.85354995727539,
  "condition": "<=",
  "ruleId": "11",
  "node": "1811"
}
]
}
```