



Universidade do Minho
Escola de Engenharia

Tiago Alberto da Silva Ribeiro

Gestão de *Workflows* Clínicos

Dissertação de Mestrado (Integrado):
Engenharia e Gestão de Sistemas de Informação

Trabalho efetuado sob a orientação do
Professor Doutor Manuel Filipe Santos
Professor Doutor António Carlos Silva Abelha

Outubro 2018

Declaração

Nome: Tiago Alberto da Silva Ribeiro

Nº Cartão Cidadão: 14621239

Tel./Telem.: 915318902

Correio eletrónico: a70064@alunos.uminho.pt / tiagosold@gmail.com

Título da dissertação: Gestão de workflows clínico

Orientador: Prof. Dr. Manuel Filipe Santos

Co-orientador: Prof. Dr. António Carlos Silva Abelha

Ano de conclusão da dissertação: 2018

Designação do Mestrado: Mestrado Integrado em Engenharia e Gestão de Sistemas de Informação

É AUTORIZADA A REPRODUÇÃO INTEGRAL DESTA DISSERTAÇÃO APENAS PARA EFEITOS DE INVESTIGAÇÃO, MEDIANTE DECLARAÇÃO ESCRITA DO INTERESSADO, QUE A TAL SE COMPROMETE.

Guimarães, 22 /10/2018

Assinatura: Tiago Alberto da Silva Ribeiro

Declaração

Nome: Tiago Alberto da Silva Ribeiro

Nº Cartão Cidadão: 14621239

Tel./Telem.: 915318902

Correio eletrónico: a70064@alunos.uminho.pt / tiagosold@gmail.com

Título da dissertação: Gestão de workflows clínico

Orientador: Prof. Dr. Manuel Filipe Santos

Co-orientador: Prof. Dr. António Carlos Silva Abelha

Ano de conclusão da dissertação: 2018

Designação do Mestrado: Mestrado Integrado em Engenharia e Gestão de Sistemas de Informação

Concordo que a minha dissertação seja colocada no repositório da Universidade do Minho com o seguinte estatuto:

Disponibilização do conjunto do trabalho para acesso exclusivo na Universidade do Minho durante o período de 3 anos, sendo que após o tempo assinalado autorizo o acesso mundial.

Guimarães, 22 /10/2018

Assinatura: Tiago Alberto da Silva Ribeiro

Agradecimentos

A conclusão desta dissertação de mestrado marca o fim de um marco importante na minha vida pessoal e académica. Porém, esta dissertação de mestrado só foi concretizável devido ao apoio de várias pessoas.

Primeiramente, gostaria de agradecer aos meus Pais, pelo esforço que sempre fizeram para garantir o meu sucesso pessoal e académico, mesmo quando as circunstâncias não eram as melhores.

À minha irmã, Filipa Ribeiro, por ter estado sempre ao meu lado quando precisei e me ter aturado ao longo destes anos.

Ao Professor Manuel Filipe Santos, meu orientador, por ter tornado o tema possível e pelos conselhos e indicações dadas ao longo da dissertação que me permitiram concluí-la com sucesso. Ao Professor Carlos Filipe Portela, pelo apoio dado, numa fase inicial da dissertação. Ao meu co-orientador António Abelha pelos esclarecimentos dados relativamente à dissertação. Ao Sérgio Oliveira, pelo apoio dado em relação ao Protótipo desenvolvido ao longo da dissertação, mostrando-se sempre disponível para esclarecer qualquer dúvida.

Um agradecimento à Universidade do Minho, por ter sido uma segunda casa ao longo destes últimos cinco anos e aos professores que tive o privilégio de conhecer neste espaço temporal, que contribuíram para o aumento do meu conhecimento em diversas matérias lecionadas e na preparação para o desenvolvimento desta dissertação de mestrado.

Aos meus colegas de curso pelo que passamos juntos durante estes anos e pelos laços criados que levarei para sempre comigo. Um especial agradecimento ao José Correia, pela sua ajuda e perspetiva nas últimas semanas em relação ao conteúdo desta dissertação. Haveria muitos mais a destacar, mas receio que tornaria este texto demasiado extenso.

Ao meu grupo de amigos, alguns de longa data, outros mais recentes, mas que sempre se mostraram presentes a ajudar-me nas minhas dificuldades e a ultrapassar os meus receios.

Por fim, um agradecimento à minha avó, que infelizmente já não se encontra entre nós, mas teve uma grande influência no meu crescimento e na pessoa que sou hoje. Tenho a certeza que este seria um momento do qual ficaria muito orgulhosa.

Resumo

Esta dissertação enquadra-se num tema de Dissertação do Mestrado em Engenharia e Gestão de Sistemas de Informação na Universidade do Minho. A investigação foi realizada tendo como âmbito o Centro Hospitalar do Porto. Esta instituição tem como missão a prestação de cuidados que melhorem a saúde dos doentes e da população, em atividades de elevada diferenciação e no apoio e articulação com as restantes instituições de saúde, incentivando a investigação como meio de desenvolvimento da ciência e tecnologia da saúde.

Diariamente, os profissionais de saúde desempenham um elevado número de processos, uns mais complexos do que outros, relacionados com os cuidados a prestar a doentes em instituições de saúde. Apesar dos sistemas de processos clínicos eletrónicos serem essenciais para a qualidade e eficiência dos serviços prestados por clínicos, a realidade é que a integração dos *workflows* clínicos nesses sistemas tem sido problemática devido à natureza complexa dos processos clínicos. Por essa razão, os gestores de instituições de saúde têm optado por manter alguns dos *workflows* clínicos em papel, o que tem comprometido a qualidade e eficiência dos serviços prestados.

O propósito deste estudo passou por investigar se o modelo *OpenEHR* poderia ser aplicado na configuração e gestão dos *workflows* clínicos, de forma a solucionar o problema da ausência de integração destes *workflows* no sistema de processo clínico eletrónico no CHP. Assim, esta dissertação endereçou este problema propondo o desenvolvimento de uma solução com o foco nos *workflows* de formulários clínicos. Esse Protótipo desenvolvido foi avaliado, de forma a perceber os contributos que o modelo *OpenEHR* pode proporcionar aos *workflows* clínicos. Antes do desenvolvimento do módulo, foi necessário recolher informação dos processos de negócio no CHP, de forma a que o Protótipo desenvolvido reflita os fluxos da organização. Esta fase permite obter uma melhor perceção dos *workflows* clínicos desempenhados pelos profissionais de saúde e, conseqüentemente, facilita o desenvolvimento dos sistemas que devem fazer parte do artefacto final produzido.

Os resultados obtidos permitem concluir que, de facto, o modelo *OpenEHR* pode ser aplicado na configuração e gestão de *workflows* clínicos, pois provou-se a interoperabilidade e flexibilidade do Protótipo desenvolvido. A configuração e gestão dos *workflows* ficou a cargo da ferramenta de *Business Process Modeling*, enquanto que a aplicação responsável por os executar reflete as modificações efetuadas nos *workflows* clínicos da ferramenta BPM.

Palavras Chave: *Workflows* clínicos, Process clínico, *Business Process Management*, Processo clínico eletrónico, *OpenEHR*

Abstract

This dissertation is framed in a Dissertation project in Engineering and Management of Information Systems master's degree at University of Minho. The research was conducted at Santo Antonio Hospital (CHP). This institution has the mission of providing care which can improve the health of the patients and population, in highly differentiated activities and in the support and articulation with other health institutions, encouraging research as a mean of developing science and technology in the Health sector.

Daily, health professionals carry out a large number of processes, some more complex than others, related with healthcare provided to patients in health institutions. Although electronic health records systems are essential to the quality and efficiency of healthcare, the integration of clinical workflows in those systems has been problematic due to the complex nature of some clinical processes. For that reason, some health institution managers have opted to maintain a few clinical workflows on paper, which has been compromising the quality and efficiency of several provided services.

The purpose of this study was to investigate if the OpenEHR model can be applied in the configuration and management of clinical workflows, so the absence of integration of some clinical workflows problem in the electronic health records system in CHP can be solved. Taking this into account, the dissertation addresses the referenced problem with the development of a solution which focus on the clinical workflows forms. This developed prototype was evaluated, so we can better understand the OpenEHR model contribution to the clinical workflows. Before developing the solution, the business processes from the CHP need to be understood so the Prototype reflects the organization flows. This phase allows us to gain a better perception on the clinical workflows carried out by health professionals and, consequently, eases the development of the database and web application, which will be part of the produced Artefact.

The results allowed to conclude that the OpenEHR model can be applied to the configuration and management of the clinical workflows, because it proved the interoperability and flexibility of the developed Prototype. The Business Process Modeling tool was responsible for the configuration and management of the workflows, while the web application, responsible for carrying them out, reflects the changes made in the clinical workflows on the BPM tool.

Keywords: *Clinical workflows, Clinical process; Business Process Management, Electronic Health Record, OpenEHR*

Índice

Agradecimentos.....	iii
Resumo.....	iv
<i>Abstract</i>	v
Índice.....	vi
Lista de Figuras.....	ix
Lista de Tabelas.....	xi
Lista de Siglas e Acrónimos.....	xii
CAPÍTULO 1. Contextualização.....	1
1.1 Enquadramento e Motivação.....	1
1.2 Objetivos e Resultados esperados.....	2
1.3 Estrutura do Documento.....	3
CAPÍTULO 2. Abordagens Metodológicas.....	6
2.1 Design Science Research.....	6
2.2 <i>Case Study</i>	8
CAPÍTULO 3. Revisão de Literatura.....	11
3.1 Estratégia de Pesquisa.....	11
3.1.1 Termos pesquisados.....	11
3.1.2 Fontes.....	11
3.1.3 Critérios de seleção.....	12
3.2 Processo Clínico.....	12
3.3 <i>Workflow</i> Clínico.....	13
3.3.1 Sistemas de Informação <i>Workflow</i>	15
3.3.2 Notação BPMN.....	17
3.4 Processo Clínico Eletrónico.....	19
3.4.1 Exemplo de aplicação – Centro Hospitalar do Porto.....	22
3.4.2 Exemplo de aplicação – Urgência Pediátrica Integrada do Porto.....	23
3.4.3 Integração de <i>Workflows</i> clínicos.....	24
3.4.4 <i>OpenEHR</i>	25

3.5	Desenvolvimento do Artefacto.....	26
3.5.1	Requisitos	26
3.5.2	Base de dados.....	28
3.5.3	Aplicação <i>Web</i>	31
3.6	Ferramentas de Modelação de Processos	33
3.7	Soluções existentes de Gestão de <i>workflows</i> clínicos e <i>OpenEHR</i>	34
3.7.1	Gestão de <i>workflows</i> clínicos e tomada de decisão	34
3.7.2	Solução baseada em arquétipos para monitorizar doentes com doenças crónicas.....	36
3.7.3	Integração de processos clínicos flexíveis utilizando a notação BPMN 2.0	37
CAPÍTULO 4.	Protótipo da aplicação de gestão de <i>workflows</i> clínicos	40
4.1	Fases da construção do Protótipo	40
4.2	Conceção do Protótipo	41
4.2.1	Definição dos requisitos do Protótipo	42
4.2.2	Lista de Ferramentas.....	42
4.2.3	Características da infraestrutura disponível	46
4.2.4	Arquitetura do Protótipo.....	46
4.3	Desenvolvimento do Protótipo.....	48
4.3.1	Base de dados.....	48
4.3.1.1	Diagrama de entidade-relacionamento.....	48
4.3.1.2	Descrição de entidades	49
4.3.2	<i>ProcessMaker</i>	53
4.3.2.1	Configurações de administrador	53
4.3.2.2	Design dos Processos	55
4.3.3	Aplicação <i>web</i> de gestão de <i>workflows</i> clínicos.....	58
4.3.3.1	Funcionalidades.....	58
4.3.3.2	Diagramas de Sequência.....	60
4.3.3.3	Interface	62
4.3.3.4	Lógica da aplicação.....	64

CAPÍTULO 5. Implementação de um workflow clínico	71
5.1. Modelação do workflow	71
5.1.1 Representação utilizando a notação BPMN 2.0	72
5.1.2 Objetos utilizados na modelação do <i>workflow</i>	73
5.2. Execução do workflow	76
5.2.1 Workflow real.....	76
5.2.2 Workflow modificado.....	81
5.3. Avaliação dos requisitos do Protótipo	82
CAPÍTULO 6. Conclusões.....	86
6.1. Análise dos Objetivos.....	86
6.2. Resultados atingidos	87
6.3. Trabalho futuro	87
Referências	89
Apêndices	94

Lista de Figuras

Figura 1 - Modelo de Processos Design Science Research	7
Figura 2 - Modelo Conceitual do workflow clínico.....	14
Figura 3 – Ciclo de vida BPM.....	17
Figura 4 - Metodologia Dois-Níveis.....	26
Figura 5 - Fases do Desenvolvimento e Gestão de requisitos	27
Figura 6 - Fases principais da criação de uma base de dados.....	30
Figura 7 - Fases da construção do Protótipo.....	41
Figura 8 - Arquitetura do Protótipo	47
Figura 9 - Diagrama Entidade-Relacionamento BD.....	49
Figura 10 - Interface Administrador ProcessMaker.....	53
Figura 11 - Configuração do Servidor de Email	54
Figura 12 - Utilizadores da Plataforma.....	54
Figura 13 - Diagrama de Sequência: Aceder à Plataforma	60
Figura 14 - Diagrama de Sequência: Iniciar Caso	61
Figura 15 - Diagrama de Sequência: Executar Workflow	62
Figura 16 - Interface "Tipo de Utilizador"	62
Figura 17 - Interface "Seleção de Caso" por parte do Utente	63
Figura 18 - Interface Login	63
Figura 19 - Interface Lista de Casos	64
Figura 20 - Interface Exemplo de formulário	64
Figura 21 - Exemplo Formulário ProcessMaker.....	68
Figura 22 - Reencaminhamento para formulário.....	69
Figura 23 - Workflow "Pedido Exame TAC"	73
Figura 24 - Objeto "Conexão à Base de Dados"	75
Figura 25 - Objecto Output Document "Informação do Caso"	75
Figura 26 - Iniciar um Caso "Processo Exame TAC"	76
Figura 27 - Formulário "Pedido de Exame TAC"	77
Figura 28 - Formulário "Verificar se exames são suficientes"	77
Figura 29 - Formulário "Recolher variáveis".....	78
Figura 30 - Técnico: Lista de Casos por responder	78

Figura 31 - Formulário "Aceitar realizar exame"	79
Figura 32 - Formulário "Publicar Resultados"	79
Figura 33 - Exame criado na Base de dados.....	80
Figura 34 - Workflow criado na Base de dados	80
Figura 35 - Documento gerado do workflow executado	80
Figura 36 - Workflow "Pedido Exame TAC" modificado	81
Figura 37 -Técnico: Caso com workflow modificado.....	81
Figura 38 - Formulário "Realizar Exame" workflow modificado.....	82
Figura 39 - Profissional de Saúde: Caso com workflow modificado.....	82
Figura 40 - Função pmRestLogin.....	94
Figura 41 - Função pmRestRequest	95
Figura 42 - Função fillLists	96
Figura 43 - Código: Iniciar Caso	96
Figura 44 - Código: Descarregar informação do workflow.....	96
Figura 45 - Código: Dynaform	97
Figura 46 - Código Route_Case.....	97
Figura 47 - Código: Gerar documento com informação do workflow	97
Figura 48 - Trigger "Verificar utente e exames".....	98
Figura 49 - Trigger "Preencher variáveis para documento"	98
Figura 50 - Trigger "Guardar workflow"	98
Figura 51 - Trigger "Dados Profissional de Saúde"	99
Figura 52 - Trigger "Dados Técnico"	99
Figura 53 - Evento "Envio de email de recusa"	99
Figura 54 - Formulário "Recolher variáveis" para Utente não existente no sistema	100
Figura 55 - Mensagem de reencaminhamento do Caso	100
Figura 56 - Formulário "Exames Suficientes" após submissão do workflow exemplo	100

Lista de Tabelas

Tabela 1 - Atividades Clínicas.....	12
Tabela 2 - Tipos de sistemas PCE	19
Tabela 3 - Benefícios potenciais sistemas PCE	21
Tabela 4 - Características Ferramentas BPM	33
Tabela 5 - Versões instaladas.....	44
Tabela 6 - Linguagens de Programação do Protótipo	45
Tabela 7 - Requisitos de ferramentas	45
Tabela 8 - Especificações da Máquina.....	46
Tabela 9 - Entidade Tipo workflow	49
Tabela 10 - Entidade Workflow	50
Tabela 11 – Entidade Tipo Exame.....	51
Tabela 12 - Entidade Utente.....	51
Tabela 13 - Entidade Exame	52
Tabela 14 - Elementos BPMN 2.0 do ProcessMaker	55
Tabela 15 - Objetos de Processo	56
Tabela 16 – Funcionalidades da aplicação de gestão de workflows clínicos	59
Tabela 17 - Métodos HTTP	66
Tabela 18 – Variáveis e Triggers por Tarefa.....	73
Tabela 19 - Avaliação dos requisitos do Protótipo	82

Lista de Siglas e Acrónimos

BD – Base de Dados

BPM – *Business Process Management*

BPMN – *Business Process Modal and Notation*

BPMS – *Business Process Management Systems*

CHP – Centro Hospitalar do Porto

DBMS – *Database Management System*

DER – Diagrama entidade-relacionamento

DSR – Design Science Research

EHR – *Electronic Health Records*

HTML – *HyperText Markup Language*

PCE – Processo Clínico Eletrónico

PHP – *HyperText Preprocessor*

REST – Representational State Transfer

SI – Sistemas de Informação

SQL – *Structured Query Language*

TAC – Tomografia Computorizada

TI – Tecnologias de Informação

UPIP - Urgência Pediátrica Integrada do Porto

WMS – *Workflow Management*

XML – *Extensible Markup Language*

CAPÍTULO 1. Contextualização

Neste capítulo é apresentado o enquadramento e motivação para a realização da investigação, os objetivos e resultados esperados e a estrutura do documento desenvolvido.

1.1 Enquadramento e Motivação

Atualmente, os registos eletrónicos são um processo essencial em qualquer área organizacional. A medicina não é exceção, por isso, tem-se assistido nos últimos anos a um aumento significativo de soluções tecnológicas para o apoio à tomada de decisão clínica e o registo eletrónico dos processos clínicos. A integração destas tecnologias obrigou à implementação de mudanças nos *workflows* clínicos existentes, o que provocou alguma desconfiança nos profissionais de saúde, visto que viram as atividades clínicas que desempenhavam diariamente a sofrerem alterações (Kilsdonk, Peute, Knijnenburg, & Jaspers, 2011).

Uma boa parte da população já passou pela necessidade de marcar de consultas em clínicas ou hospitais públicos, seja com médicos de família ou especialidades. Essa experiência raramente é positiva, pois os tempos de espera são, por norma, bastante longos, devido à grande afluência de utentes aos serviços de saúde e a falta de capacidade e meios para responder aos pedidos. Tendo isto em conta, a utilização conjunta dos *workflows* clínicos e o sistema de processo clínico eletrónico é visto como uma das possíveis soluções para a diminuição dos tempos de espera, tanto em consultas como no processo de atendimento, pois permite que os utentes possuam um maior controlo e informação do seu estado clínico e tratamentos a aplicar, não tendo a mesma necessidade de se descolarem aos serviços de saúde. Esta integração também beneficia os Profissionais de Saúde, ao permitir que o PCE passe a englobar os processos clínicos desempenhados diariamente, o que possibilita a agilização do atendimento clínico e cruzamento de informação entre clínicos.

O Centro Hospitalar do Porto centra-se na prestação de cuidados que melhorem a saúde dos doentes e da população, em atividades de elevada diferenciação e no apoio e articulação com as restantes instituições de saúde, destacando-se pelas boas práticas clínicas e de gestão.

Os *workflows* de trabalhos clínicos associados à atuação clínica, por vezes, não estão contemplados no sistema PCE, tendo como consequências a perda de controlo, por parte dos clínicos do CHP, sobre os seus utentes e os cuidados a prestar. Estes problemas podem comprometer

seriamente a eficiência e a qualidade dos serviços prestados e, se o CHP se apresenta como uma referência na área das boas práticas clínicas e de gestão, então é de esperar que o hospital esteja sempre na vanguarda tecnológica. Na última década, a área da Medicina tem evoluído de forma suportar a interoperabilidade de dados eletrónicos, logo espera-se que o CHP evolua nesse sentido de forma a ser capaz de continuar a denominar-se como uma referência na sua área de atuação. A dificuldade dos centros hospitalares de saúde em mantarem-se tecnologicamente atualizados baseia-se no facto de possuírem poucos meios para investir nas soluções tecnológicas que poderiam melhorar os seus processos e, conseqüentemente, a atuação clínica (Pearce & Bainbridge, 2014).

O artefacto desenvolvido nesta dissertação visa abordar os problemas de integração dos *workflows* clínicos no sistema PCE do CHP, de forma a melhorar o controlo e qualidade dos processos desempenhados diariamente pelos profissionais de saúde. Dito isto, a ideia da investigação passa por perceber se é possível a aplicação do modelo *OpenEHR* na configuração e gestão dos *workflows* clínicos. De referir que o foco do artefacto esteve nos *workflows* de formulários clínicos que utilizam arquétipos baseados em *OpenEHR*.

A principal motivação para a realização desta dissertação, incide nos contributos e benefícios que a investigação pode oferecer ao setor da Saúde, numa altura em que a área tem sido particularmente escrutinada, em consequência da crescente preocupação da população, indignada com a falta de meios dos centros hospitalares para prestar cuidados básicos de saúde (Correia, 2011). De notar que a formalização dos *workflows* possibilita o aumento da eficácia e, conseqüentemente, da qualidade dos serviços prestados, devido à padronização das decisões e atuações clínicas. O contributo da investigação para os Sistemas de Informação não deve ser menosprezado, pois permitirá, caso os objetivos sejam atingidos, demonstrar a importância que os SI podem ter na melhoria dos fluxos e processos no setor da Saúde.

1.2 Objetivos e Resultados esperados

O principal objetivo desta dissertação é investigar se o modelo *OpenEHR* pode ser aplicado na configuração e gestão dos *workflows* clínicos. O foco dos *workflows* clínicos nesta investigação esteve nos formulários baseados em arquétipos do módulo *OpenEHR*.

De forma a avaliar se o modelo *OpenEHR* poderia ser aplicado como descrito no objetivo da investigação, foi necessário criar um Protótipo que garantisse a interoperabilidade e flexibilidade entre a aplicação de execução dos *workflows* e a ferramenta BPM que permite a sua configuração e gestão. A

questão da interoperabilidade possibilita a comunicação entre os sistemas, o que significa que a aplicação pode ser gerida pelas configurações presentes na ferramenta BPM, o que permite lidar com a situação atual dos *workflows* clínicos não estarem contemplados no sistema PCE. A flexibilidade aborda outra necessidade no Centro Hospitalar do Porto, pois, atualmente, qualquer alteração nos processos é dificultada por obrigar a modificações no código base. Devido à constante mutação dos processos nas organizações, a utilização de uma ferramenta BPM facilita o processo de alteração dos *workflows*, devido à sua natureza mais gráfica e, em conjunto com a interoperabilidade, deve permitir que as modificações sejam refletidas na execução dos *workflows*. Devido a estas questões abordadas, nota-se a importância que o Protótipo pode assumir para a gestão dos *workflows* clínicos no CHP.

A investigação foi desenvolvida com apoio do Centro Hospitalar do Porto. Os dados utilizados neste projeto provenientes do CHP, foram regidos por critérios de confidencialidade, significando que apenas podem ser aplicados no âmbito da dissertação.

Os objetivos específicos da dissertação são os seguintes:

- Levantamento de ferramentas BPM;
- Recolha e modelação dos processos clínicos;
- Levantamento e definição dos requisitos do Protótipo a ser desenvolvido;
- Construção do Protótipo;
- Avaliação dos resultados.

Os resultados esperados provenientes do desenvolvimento desta dissertação incluem os *workflows* clínicos modelados, o desenvolvimento de um Protótipo que garanta a interoperabilidade entre a aplicação de gestão dos *workflows* clínicos e a ferramenta BPM e assegure que, caso ocorram modificações nos *workflows* dessa ferramenta, então o Protótipo desenvolvido reflita essas mudanças. Após o desenvolvimento do Protótipo, espera-se que a avaliação dos seus resultados possibilite responder à questão de investigação “*É possível a aplicação do modelo OpenEHR na configuração e gestão dos workflows clínicos?*”.

1.3 Estrutura do Documento

Este documento encontra-se estruturado em seis capítulos, cujos conteúdos principais são:

- **Contextualização** – Neste capítulo é apresentado ao leitor o enquadramento e motivação para a realização da dissertação sobre o tema em estudo. Também contém os objetivos e resultados esperados do projeto e a estrutura do documento do projeto de dissertação.
- **Abordagens Metodológicas** – Capítulo em que se apresenta ao leitor as abordagens metodológicas a utilizar na dissertação. Nesta secção está presente um resumo da metodologia *Design Science Research* e *Case Study*, utilizadas para o desenvolvimento do artefacto e análise dos processos clínicos do CHP, respetivamente.
- **Revisão de Literatura** – Capítulo que começa por apresentar as estratégias de pesquisa utilizadas, desde termos pesquisados, fontes e critérios na seleção de informação. Após a primeira subsecção, encontra-se o estado da arte dos conceitos relevantes para a dissertação. Os conceitos explorados são: Processo clínico, *Workflows* clínicos e Processo clínico eletrónico. Neste capítulo ainda se aborda o desenvolvimento do módulo *OpenEHR*, mais concretamente, o papel dos requisitos, base de dados e aplicação *web* no artefacto. Por fim, estão presentes um conjunto de ferramentas *open-source* que podem ser utilizadas na fase da modelação de processos de negócio.
- **Protótipo da aplicação de gestão de *workflows* clínicos** – Neste capítulo encontra-se toda a informação relativa ao desenvolvimento do Protótipo. Numa fase inicial, é possível visualizar as fases do desenvolvimento do Protótipo, os requisitos definidos e a arquitetura desenvolvida. De seguida é apresentada a informação relativa à ferramenta BPM escolhida, desde as razões para a seleção dessa plataforma e algumas observações em relação ao funcionamento do *software*. Para finalizar o capítulo, é apresentada uma explicação detalhada da aplicação gestão de *workflows* clínicos desenvolvida, começando por diagramas de sequência que explicam as trocas de mensagens entre a aplicação *web* desenvolvida e ferramenta BPM do Protótipo, as *interfaces* criadas com as quais o utilizador pode interagir e o código criado que permite o funcionamento lógica da aplicação.
- **Implementação de um *workflow* clínico** – De forma a testar se o Protótipo cumpriu com os objetivos delineados para esta dissertação, foi necessário implementar um *workflow* de formulários clínico real do CHP. Inicialmente, encontra-se uma explicação do *workflow* utilizado e a modelação do mesmo na ferramenta BPM selecionada e, de seguida, uma explicação dos objetos que foram necessários criar na plataforma para suportar a execução do *workflow*. Logo após, apresenta-se a execução do *workflow* clínico, com o intuito de comprovar a utilidade e

flexibilidade do Protótipo desenvolvido e, por fim, uma avaliação se os requisitos do Protótipo foram cumpridos.

- **Conclusões** – Neste capítulo pode-se observar, numa fase inicial, a análise dos objetivos delineados para a investigação. De seguida, encontra-se a análise dos resultados atingidos ao longo da investigação desenvolvida. Por fim, é feita uma reflexão sobre o trabalho futuro a efetuar, no seguimento do artefacto desenvolvido nesta dissertação.

CAPÍTULO 2. Abordagens Metodológicas

No desenvolvimento da dissertação foram utilizadas duas metodologias. A *Design Science Research* enquanto metodologia de investigação e construção do artefacto e o *Case Study* para o estudo detalhado dos *workflows* clínicos existentes no Centro Hospitalar do Porto. Para uma correta aplicação destas metodologias, as suas fases deverão ser respeitadas ao longo do projeto.

2.1 Design Science Research

Segundo Vaishnavi & Kuechler (2008), a *Design Science Research* tem como objetivo a melhoria da compreensão de aspetos comportamentais em Sistemas de Informação. Para isso contém duas atividades principais, “*criação de novo conhecimento através do design de artefactos originais ou inovadores*” e “*análise do uso e/ou desempenho do artefacto*” (Vaishnavi & Kuechler, 2008). Para Hevner et al. (2004), *design science* é um processo de resolução de problemas e “*o conhecimento e compreensão de um problema e a sua solução são adquiridos na construção e aplicação de um artefacto*”.

Hevner et al. (2004) criaram um conjunto de *guidelines* para aplicar à *Design Science Research* em Sistemas de Informação. Segundo Hevner et al. (2004), “*a DSR requer a criação de um artefacto inovador e relevante para abordar um determinado problema*”. A avaliação do artefacto é fundamental para verificar se o que está a ser desenvolvido é relevante para solucionar o problema em estudo. O artefacto deve ajudar a “*solucionar um problema para o qual ainda não existe solução ou solucionar um problema conhecido de uma forma mais eficiente e efetiva*” (Hevner et al., 2004). Hevner et al. (2004) referem que “*o artefacto deve ser rigorosamente, coerentemente e consistentemente definido*”. O desenvolvimento do artefacto deve utilizar teorias existentes e recolher conhecimento que delas advém para encontrar uma solução para o problema. Os resultados da pesquisa realizada devem ser comunicados a quem os poderá utilizar e a um conjunto de investigadores que deve investigar o contexto desses resultados (Hevner et al., 2004).

Vaishnavi & Kuechler (2008) desenvolveram um modelo de processos para a DSR, representado na Figura 1, que vai de encontro com as *guidelines* criadas por Hevner et al. (2004).

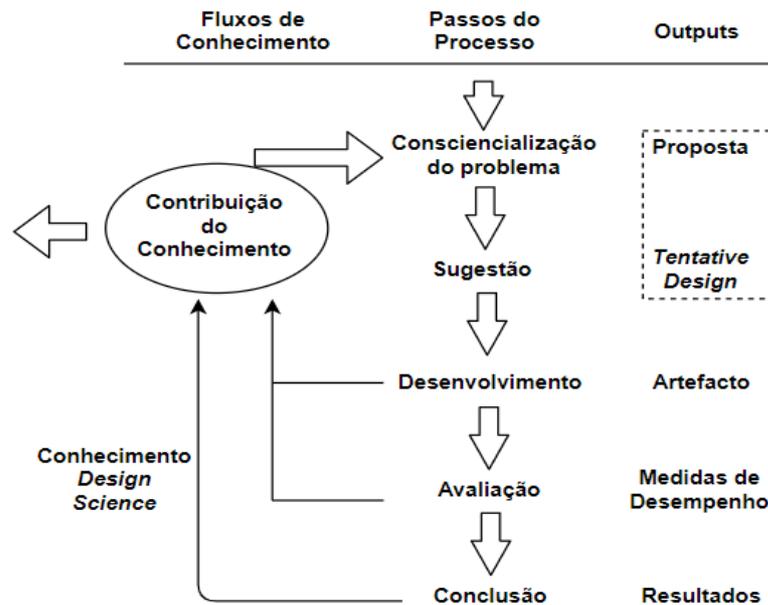


Figura 1 - Modelo de Processos Design Science Research (Adaptado de Vaishnavi & Kuechler (2008))

O Modelo de Processos da DSR, visível na Figura 1, apresenta o conjunto de fases da metodologia e o *output* de cada uma delas. O objetivo da metodologia é a criação de um artefacto inovador e capaz de solucionar um problema em estudo. Após o desenvolvimento do artefacto, o sucesso do mesmo vai depender da avaliação da sua contribuição para a área em estudo e, se o artefacto desenvolvido é uma solução para o problema analisado.

Consciencialização do problema: Segundo Vaishnavi & Kuechler (2008), a consciencialização de um problema pode vir de novas descobertas nas indústrias ou da identificação de problemas ou limitações numa determinada área. Esta fase deve produzir como *output* uma proposta para a investigação do problema identificado. No contexto desta investigação, o problema identificado está relacionado com o facto dos *workflows* clínicos nem sempre estarem contemplados no processo clínico eletrónico, o que provoca constrangimentos ao nível do controlo de processos por parte dos profissionais de saúde e compromete a eficiência e qualidade dos serviços prestados no Centro Hospitalar do Porto. Nesta fase também entra o *Case Study* descrito nesta capítulo, que permitiu realizar um levantamento rigoroso do estado atual dos processos clínicos e como estão integrados no sistema de processo clínico eletrónico e os requisitos necessários para atingir o estado desejado do Protótipo.

Sugestão: Fase integral da proposta desenvolvida. É o passo criativo do processo, em que serão definidos os objetivos e as funcionalidades da investigação de acordo com elementos já existentes ou novos na literatura (Vaishnavi & Kuechler, 2008). A sugestão definida para abordar o problema desta investigação

centra-se na integração de uma *Business Process Management* na plataforma PCE do Centro Hospitalar do Porto. Essa integração implicou o desenvolvimento de um Protótipo, que permitiu visualizar e executar os *workflows* criados na BPM.

Desenvolvimento: A sugestão é, nesta fase, desenvolvida e implementada. As técnicas que serão utilizadas irão depender do artefacto a ser desenvolvido (Vaishnavi & Kuechler, 2008). Nesta investigação, o Protótipo foi desenvolvido, de acordo com os requisitos levantados e, principalmente, de forma a assegurar a interoperabilidade e flexibilidade entre a aplicação *web* e ferramenta BPM.

Avaliação: Após o desenvolvimento do artefacto, este é avaliado de acordo com os critérios desenvolvidos na primeira fase. Caso existam diferenças em relação aos critérios definidos, as modificações devem ser devidamente explicadas e justificadas (Vaishnavi & Kuechler, 2008). Nesta fase também será avaliada a utilidade, qualidade e eficácia do artefacto para o problema em investigação (Hevner et al., 2004). Quando o desenvolvimento do Protótipo estiver finalizado, este deverá ser avaliado de acordo com os requisitos levantados e as funcionalidades definidas. Caso alguma funcionalidade não esteja implementada ou tenha sido implementada de uma forma diferente da definida, as razões para as modificações efetuadas devem ser explicadas.

Conclusão: Para Hevner et al. (2004), a comunicação e apresentação do artefacto a quem irá utilizar o próprio artefacto é essencial para definir as contribuições da investigação para a indústria. Os resultados da investigação serão analisados e interpretados, representando o fim do ciclo do modelo de processos da DSR (Vaishnavi & Kuechler, 2008). No contexto desta investigação, o Protótipo desenvolvido foi apresentado a um conjunto de elementos da equipa de desenvolvimento do sistema PCE que avaliaram a sua capacidade em suprimir a maioria dos problemas e limitações presentes no Centro Hospitalar do Porto em relação aos *workflows* clínicos. A escrita de artigos científicos referentes ao estudo permite a partilha do conhecimento adquirido com a comunidade científica da área em investigação.

2.2 *Case Study*

Segundo Gillham (2000), um caso pode ser um indivíduo, um grupo, uma comunidade ou uma instituição se forem um caso singular. Também é possível investigar casos múltiplos, como por exemplo, múltiplas instituições ou múltiplos indivíduos, dependendo dos objetivos da investigação. Para Gillham (2000), um *case study* investiga um dos casos acima descritos para responder a questões específicas de pesquisa. A investigação deve ser apoiada com um conjunto de provas distintas de forma a obter as melhores respostas às perguntas levantadas pela investigação. Segundo Yin (2009), o *case study* deve

ser usado para estudar um fenómeno dentro do seu contexto real e se esse fenómeno é pertinente para a investigação.

Para Yin (2009), existem quatro componentes essenciais nos casos de estudo:

Questões do estudo: A definição das questões permite clarificar a essência do estudo (Yin, 2009). Nesta investigação pretende-se avaliar os *workflows* relacionados com a atuação clínica, principalmente os que não são contemplados no processo clínico eletrónico, logo, a questão em estudo é a seguinte “Como integrar os *workflows* clínicos no processo clínico eletrónico, de forma a melhorar os serviços prestados pelos profissionais de saúde?”.

Proposições do estudo: Para Yin (2009), “*cada proposição direciona a atenção para algo que deve ser analisado dentro dos limites do estudo*”. As proposições podem ser necessárias quando as questões definidas são vagas e não direcionam o investigador para o assunto a ser estudado. Alguns *case studies* poderão não ter proposições quando a questão de estudo é o assunto a ser explorado (Yin, 2009). No contexto desta investigação, a questão de estudo definida não é suficientemente esclarecedora das análises que deverão ser efetuadas. De seguida apresentam-se as proposições definidas para o projeto:

- **1ª Proposição: Alguns *workflows* clínicos estão presentes no processo clínico eletrónico.** Esta proposição permite analisar se existem *workflows* clínicos já integrados no processo clínico eletrónico, caso a resposta seja positiva, enumerá-los e verificar se existem melhorias que podem ser implementadas nesses *workflows*. Mas o grande objetivo desta proposição, é identificar os *workflows* clínicos que faltam integrar no PCE, de forma a definir e modelar os processos *Business Process Management* ao nível da atuação clínica.

- **2ª Proposição: A integração dos *workflows* clínicos nos processos clínicos eletrónicos melhoram os serviços prestados pelos profissionais de saúde.** A opinião dos profissionais de saúde em relação ao estado dos *workflows* clínicos, limitações atuais e melhorias a serem implementadas, é essencial para o desenvolvimento deste estudo e do artefacto final. O artefacto deve ser capaz de suportar as necessidades dos clínicos e, conseqüentemente, melhorar os serviços prestados.

Unidade de análise: A unidade de análise responde à questão “*o que está a ser estudado?*”, pode ser, por exemplo, um indivíduo, uma instituição ou uma comunidade. Quando se definem as unidades de análise, também se deve definir os dados que serão recolhidos dessa mesma unidade, para que se possa estabelecer uma relação com as proposições definidas (Yin, 2009). Neste projeto, foi investigado o contexto do CHP. Os elementos de contacto do sistema PCE também participaram neste estudo de forma a recolher informação em relação aos *workflows* clínicos do CHP.

Ligação dos dados às proposições: Segundo Yin (2009), nesta fase deverão ser estabelecidos padrões entre os dados recolhidos e as proposições definidas anteriormente, para que se possam tirar conclusões da investigação realizada. Nesta fase será feita a análise dos dados recolhidos e da sua relevância em relação às proposições definidas, para que numa fase posterior se possam definir os processos BPM a serem modelados e os requisitos do artefacto a ser desenvolvido.

CAPÍTULO 3. Revisão de Literatura

Este capítulo contém as estratégias utilizadas no processo de investigação de informação para a dissertação. O estado da arte também se encontra presente neste capítulo e engloba os conceitos considerados relevantes para o trabalho de dissertação desenvolvido. Por fim, apresenta-se uma lista de ferramentas *open-source* que podem ser úteis na modelação dos processos de negócio ao nível hospitalar.

3.1 Estratégia de Pesquisa

Nesta secção foi realizada uma descrição dos termos pesquisados na revisão de literatura desta dissertação, as fontes utilizadas para aceder à informação e os critérios utilizados para seleccionar essa informação.

3.1.1 Termos pesquisados

As pesquisas efetuadas incidiram sobre os seguintes conceitos: *Business Process Management*, *BPM modelling tools*, *Clinical workflows*, *Clinical process*, *Database*, *Electronic Health Records*, *EHR systems*, *Information Systems Workflows*, *OpenEHR*, *Software Requirements* e *Web application*.

De referir que, em algumas pesquisas foi necessário combinar alguns dos termos acima mencionados, de forma a obter informação mais relevante para a dissertação.

3.1.2 Fontes

Os seguintes repositórios foram utilizados para as pesquisas efetuadas:

- *Association for Information Systems*;
- *Google Scholar*;
- *IEEE*;
- *ISI Web of Knowledge*;
- *ScienceDirect*;
- *Scopus*;
- *SpringerLink*.

3.1.3 Critérios de seleção

Os principais critérios utilizados na seleção de artigos foram uma combinação da relevância do artigo/obra para o tema da dissertação, o número de citações e a data da publicação, sendo que foi dada prioridade aos artigos publicados nos últimos cinco anos.

3.2 Processo Clínico

Um processo clínico pode ser visto como um fluxo de trabalho específico que engloba atividades ou eventos médicos e não médicos. As atividades médicas englobam, entre outras, tratamentos, administração de medicamentos e exames médicos enquanto que as atividades não médicas estão relacionadas com a inscrição de utentes e preenchimento de formulários (Ruffolo, Manna, Cozza, & Ursino, 2007). Segundo Ruffolo et al. (2007), os processos clínicos podem ser muito complexos devido às atividades de alto risco que necessitam de ser desempenhadas nos cuidados de alguns doentes. O desenho e implementação de processos em hospitais ou clínicas permite aumentar a eficácia, eficiência e qualidade dos cuidados prestados nessas organizações. A eficácia está relacionada com os resultados ou efeitos de um tratamento, a eficiência com os tempos e os custos dos cuidados prestados e a qualidade com o sucesso dos tratamentos receitados, a percentagem de ocorrências indesejadas e satisfação do utente (Rohner, 2012).

Lee & Shartzter (2005) dividem as atividades das organizações de saúde em administrativas e clínicas, como se pode observar na Tabela 1.

Tabela 1 - Atividades Clínicas (Adaptado de Lee & Shartzter (2005))

Tipo	Atividades
<i>Administrativa</i>	<ul style="list-style-type: none">● Marcar consultas;● Documentar informação dos utentes;● Recolher e guardar os registos médicos dos doentes;● Despesas dos processos;● Reclamações;● Comunicação.
<i>Clínica</i>	<ul style="list-style-type: none">● Tratamento médico (triagem);● Histórico médico dos utentes;● Examinar e diagnosticar utentes;● Desenvolver planos de tratamento;● Educar os doentes sobre cuidados a ter ou informação que necessitam para algum tratamento;● Receitar medicação;

Tipo	Atividades
	<ul style="list-style-type: none"> • Receitar exames (vacinações, raios-x, etc) ou testes para laboratório; • Seguir tratamento dos utentes.

As atividades administrativas não fazem parte do trabalho médico, logo devem ser realizadas pelo *staff* administrativo ou recursos humanos nas organizações de saúde, como por exemplo rececionistas ou secretárias. As atividades clínicas englobam apenas o trabalho médico a desempenhar nas instituições e são exclusivamente praticados por profissionais de saúde (Lee & Shartzter, 2005).

3.3 *Workflow* Clínico

Plataformas tecnológicas têm sido usadas em várias áreas com o objetivo de suportar, agilizar processos e facilitar a gestão dos utilizadores. Dito isto, a área da Saúde não é exceção a este crescimento de soluções tecnológicas, sendo natural o surgimento de pedidos de plataformas que suportem o apoio à tomada de decisão por parte dos clínicos, facilitem o armazenamento eletrónico de dados dos utentes e a interação entre clínicos e de clínicos para utentes. A necessidade destas plataformas conduziu ao desenvolvimento de soluções que permitem suportar os serviços prestados por hospitais ou clínicas de forma a garantir a melhoria da eficiência e qualidade desses mesmos serviço (Militello et al., 2014).

Workflows clínicos são o conjunto de passos de um processo clínico que podem ser realizados por múltiplas pessoas, clínicos e utentes por exemplo, e espera-se que consumam, produzam, transformem ou troquem informação (Militello et al., 2014). O *workflow* clínico também é definido como a alocação de múltiplas tarefas realizadas por um ou vários clínicos nos processos de cuidados de saúde e na forma como os clínicos colaboram, podendo ser separados em quatro categorias: (1) estruturação de tarefas clínicas, (2) coordenação do desempenho de tarefas, (3) habilitar o fluxo de informação para suportar o desempenho de tarefas e (4) a monitorização do desempenho de tarefas (Niazkhani, Pirnejad, Berg, & Aarts, 2009). Niazkhani et al. (2009) decidiram incorporar estes aspetos num modelo, representado na Figura 2.

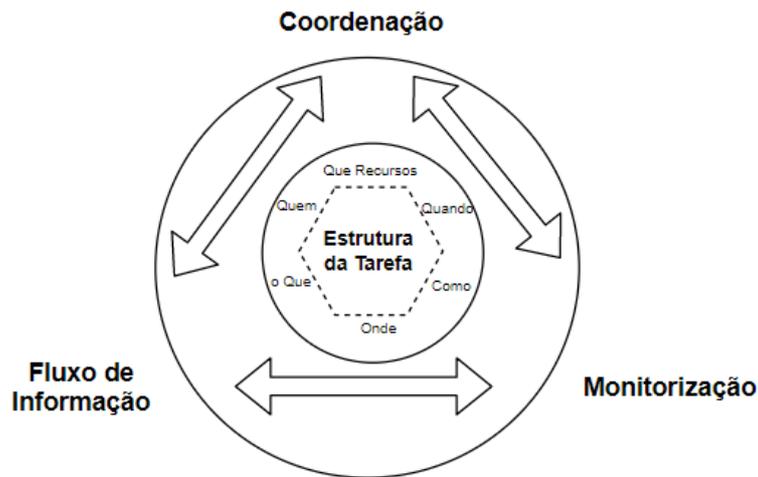


Figura 2 - Modelo Conceitual do *workflow* clínico (Retirado de Niazkhani et al. (2009))

O Modelo Conceitual do *workflow* clínico, presente na Figura 2, representa o fluxo do trabalho clínico. O trabalho clínico necessita de um conjunto de tarefas médicas para ser executado. A estruturação dessas tarefas inclui, a definição dos recursos necessários, quando/como/onde é executada a tarefa, quem a vai executar e se o que é executado está relacionado com outras tarefas ou organizações de saúde. A coordenação implica que os profissionais de saúde devem orientar o trabalho médico, desde a coordenação com os colegas, os recursos necessários, quando e quem necessita de utilizar os recursos e calendarização do trabalho clínico. Esta coordenação surge das regras, planos e normas existentes na organização, que devem ser seguidas pelos clínicos. A recolha da informação dos doentes é vital para o trabalho médico, mas como a recolha de dados pode partir de profissionais de saúde diferentes, é necessário comunicar a informação processada para permitir a colaboração entre os diferentes clínicos envolvidos. A monitorização de mudanças nos requisitos das tarefas permite que os profissionais de saúde ajustem as ações individuais e cooperativas e que supervisionem e controlem a execução das tarefas (Niazkhani et al., 2009).

Para Combi et al. (2009), os *workflows* assumem uma importância particular, pois permitem que os profissionais de saúde e administrativos se foquem em aspetos cruciais como a avaliação da qualidade dos serviços de saúde prestados, sugestões de cuidados de saúde mais adequados para os utentes e controlo do orçamento para cada atividade clínica.

Os *workflows* são cada vez mais uma preocupação para os clínicos devido à introdução de novas tecnologias de informação para os sistemas de saúde dedicados às práticas clínicas. Apesar destas novas tecnologias prometerem vários benefícios na melhoria da qualidade e eficiência dos cuidados de saúde,

a realidade é que também podem ser bastante problemáticas para os fluxos existentes nas organizações (Ash et al., 2007).

Atualmente, a necessidade de pensar no desenho dos fluxos é mais urgente e complexa devido aos seguintes fatores (Cain & Haque, 2008):

- Introdução de novas tecnologias e metodologias de tratamento nos cuidados clínicos;
- O desafio na coordenação de cuidados para os doentes crónicos;
- Cada vez mais profissionais de saúde participam nos cuidados de um doente, logo é necessário definir os novos papéis a serem desempenhados;
- Pressões para melhorar o fluxo do utente, de forma a reduzir custos e melhorar a eficiência dos cuidados médicos;
- Iniciativas para garantir a segurança dos utentes;
- Implementação de mudanças para tornar a equipa de clínicos mais orientada para o doente.

3.3.1 Sistemas de Informação *Workflow*

Segundo o *Workflow Management Coalition* (WfMC, 1999), *Workflow* é a automação total ou parte de um processo de negócio, que envolve a passagem de documentos, informação ou tarefas de um participante para outro de acordo com um conjunto de regras definidas. Um processo de negócio consiste numa sequência de atividades e possui *inputs* e *outputs* distintos, servindo um propósito dentro de uma organização ou entre organizações. Uma atividade é um passo do processo realizado por uma máquina ou um agente humano. Para Stohr & Zhao (2001), um *workflow* é um tipo de processo específico em que as transições entre as atividades são controladas por um sistema de informação. Segundo Stohr & Zhao (2001), não é necessário que a informação esteja presente em formato eletrónico.

Apesar do grande interesse das organizações em estruturas orientadas a processos, a sua implementação tem-se provado difícil para várias empresas (Davenport & Beers, 1995). A existência de infraestruturas para as tecnologias de informação, que suportam as organizações funcionais e dificultam a transição para estruturas orientadas a processos é uma das principais dificuldades sentidas nas organizações (Luftman, 2003).

Workflow Management Systems (também conhecidos como *Business Process Management Systems*) abordam as dificuldades sentidas pelas organizações, suportando a execução dos processos de negócio através da automatização coordenada de atividades e recursos de acordo com um modelo definido do processo de negócio (Georgakopoulos, Hornick, & Sheth, 1995). Na literatura mais recente,

Van Der Aalst et al. (2016) alegam que a BPM é, na realidade, uma evolução do conceito de *Workflow Management*. Enquanto o WM se foca, primariamente, na automação dos processos de negócio, a BPM tem uma abrangência mais elevada, desde a automação e análise dos processos até à gestão de operações e a organização do trabalho. A utilização de BPM para melhorar os processos de negócio não implica, necessariamente, a utilização de novas tecnologias. A modelação de processos de negócio e a sua análise através de simulações podem ajudar os Gestores de uma organização a reduzir custos enquanto melhoram os níveis do serviço (Van Der Aalst et al., 2016).

Segundo Van Der Aalst et al. (2016), um processo pode ser considerado melhorado quando contribui, de forma mais eficiente e eficaz, para o cumprimento dos objetivos estratégicos de uma organização. Os indicadores de desempenho podem ser usados para medir o impacto dos processos de negócio. Os indicadores são quantidades que podem ser determinadas para um determinado processo, assumindo que os dados necessários para calcular as métricas de desempenho estão disponíveis (Van Der Aalst et al., 2016). Dumas et al. (2013) descrevem alguns desses indicadores como dimensões de desempenho, que incluem o tempo, qualidade, custo e flexibilidade. O tempo pode ser medido através de tempo de ciclo e tempo de espera, a qualidade com a satisfação dos clientes, taxa de erros e violações ao nível de serviço e o custo através do custo por utilização, utilização de recursos e desperdício. Alguns destes indicadores de desempenho mencionados podem ser medidos facilmente como o tempo de ciclo ou tempo de espera, mas há outros que implicam mais trabalho, como o indicador de satisfação dos clientes, devido à necessidade de agregar a informação das experiências dos clientes, avaliação dos produtos e análises de lealdade (Dumas et al., 2013).

Para Dumas et al. (2013), *Business Process Management* é “*um conjunto de princípios, métodos e ferramentas para gerir processos de negócio com o objetivo de os melhorar*”. O ciclo de vida BPM pode ser visualizado na Figura 3.

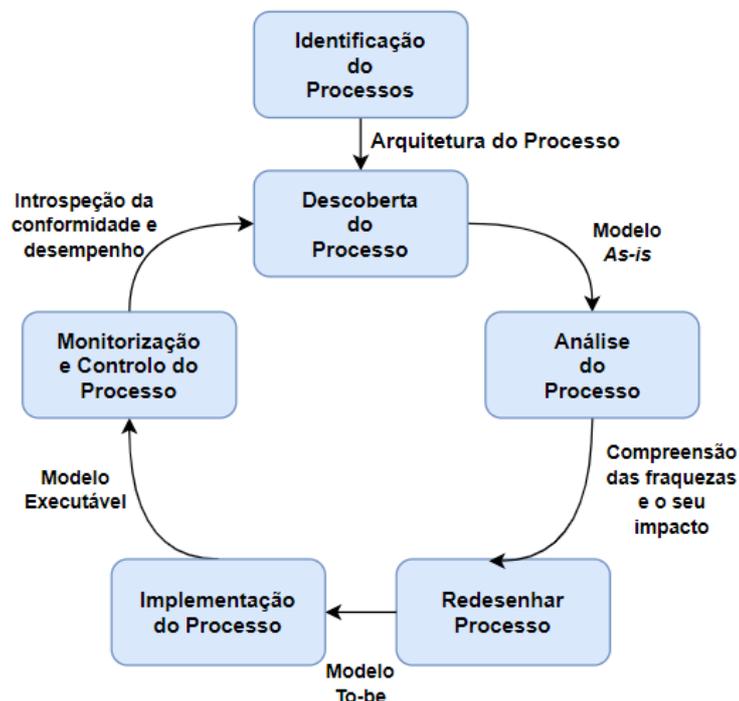


Figura 3 – Ciclo de vida BPM (Retirado de Dumas et al. (2013))

O Ciclo de Vida BPM (Figura 3) começa com a identificação do processo. Nesta fase estudam-se os limites, inter-relações do processo e como deve ser priorizado. O passo seguinte é a compreensão do modelo do processo de negócio *as is*, ou seja, a forma atual do processo necessária para realizar a sua análise. A análise do processo engloba a aplicação de um conjunto de técnicas que permite analisar o seu desempenho. A análise dos resultados da fase anterior, permite a identificação e avaliação de melhorias a desenvolver no processo, que levarão à implementação do modelo *to be*. Findada a implementação, é necessário desenvolver mecanismos e técnicas de monitorização e controlo do modelo *to be* para avaliar se o processo está a cumprir as metas de desempenho definidas (Dumas et al., 2013).

3.3.2 Notação BPMN

Em 2004, surgiu a *Business Process Modeling Notation* (BPMN) com o objetivo de normalizar a linguagem de modelação de processos de negócio. Desde essa altura, a BPMN tem sido globalmente apoiada pela comunidade académica e indústria. Após a primeira grande revisão dos elementos BPMN, a *Object Management Group*, uma organização que pretende normalizar as aplicações orientadas a objetos, lançou a versão BPMN 2.0 em 2011 (Aagesen & Krogstie, 2015).

Segundo Aagesen & Krogstie (2015), a BPMN é uma notação facilmente compreendida por todos os utilizadores do negócio, desde os analistas que criam o *draft* inicial dos processos, até aos

técnicos responsáveis por implementar a tecnologia que suporta o desempenho desses processos e, finalmente, os gestores responsáveis por gerir e monitorizar o seu desempenho. Outro fator que levou à criação da notação está relacionado com a necessidade de criar uma ponte entre a notação visual e as linguagens de execução, de forma a facilitar a perceção dos gestores do processo sobre a evolução e desempenho dos processos que desenvolveram.

A BPMN permite a criação de processos de negócio *end-to-end*, ou seja, abrangem todo o trabalho que deve ser realizado para atingir o objetivo do processo. Esta notação está projetada para cobrir vários tipos de tarefas modeladas restritas a processos de negócio. Os elementos de estruturação BPMN permitem que a pessoa que visualiza o processo possa diferenciar entre as secções do diagrama BPMN através da utilização de grupos, *pools* ou *lanes*. Existem dois tipos básicos de submodelos dentro de um modelo BPMN, que se designam por processos de negócio privados e processos públicos. Os processos de negócios privados são os processos internos numa organização e são os tipos de processo normalmente designados por *workflow* ou processo BPM. Os processos públicos retratam as interações entre duas ou mais entidades de negócio. Estas interações são definidas como uma sequência de atividades que representam os padrões de troca de mensagens entre as entidades envolvidas (Aagesen & Krogstie, 2015).

Silver (2011) definiu três níveis de uso da notação BPMN, devido ao elevado número de conceitos que foram surgindo relativos à notação. O primeiro nível designa-se modelação descritiva e é orientada para a documentação do fluxo do processo. É utilizada primariamente na comunicação entre os modelos *as-is* e *to-be* e também na implementação manual. A maior parte do uso do BPMN é a este nível. O segundo nível, modelação analítica, possibilita a modelação mais precisa em relação às exceções e eventos. Permite o suporte a análises qualitativas, quantitativas e de indicadores de desempenho. O terceiro nível denomina-se modelação executável e contém os modelos gráficos que podem ser transformados em especificações baseados em XML que acionam os mecanismos do processo. Também tornam possível o suporte à ativação automática dos modelos.

As construções da linguagem BPMN podem ser divididas em quatro categorias, os objetos de fluxo, objetos de ligação, *swimlanes* e artefactos. Os objetos de fluxo contêm as atividades, eventos e *gateways*. As atividades dividem-se em processo, subprocesso e tarefas, denotando o trabalho que é efetuado na organização. Os eventos definem-se como alguma coisa que acontece num processo e como o processo responde a isso ou como o processo gera um sinal de que algo aconteceu. Os eventos dividem-se em eventos de início, eventos intermédios e eventos de fim. Os *gateways* são usados para determinar a separação, fusão, deslocação ou união dos caminhos no processo. Os objetos de ligação

são usados para ligar os objetos de fluxo. Estes objetos podem ser do tipo fluxo de sequência, quando definem a ordem de execução das atividades num processo e fluxos de mensagem para indicar a troca de mensagens entre entidades do negócio ou cargos preparados para os enviar e receber. As *swimlanes* são utilizadas para denotar um participante num processo e funciona como um recipiente gráfico para um conjunto de atividades atribuídos a esse participante. Os artefactos podem ser objetos de dados, grupos ou anotações. Os objetos de dados oferecem informação dos recursos requeridos ou produzidos pelas atividades. Os grupos são uma ajuda visual utilizada para propósitos de documentação ou análise. As anotações de texto permitem adicionar informação sobre certos aspetos do modelo (Aagesen & Krogstie, 2015).

3.4 Processo Clínico Eletrónico

Para Häyrynen & Nykänen (2008), um sistema de processo clínico eletrónico é um repositório de informação de utentes armazenada em formato eletrónico. Essa informação pode ser trocada se estiverem garantidos os mecanismos de segurança necessários para que apenas os utilizadores autorizados a possam visualizar. O principal propósito de um sistema PCE é suportar a integração contínua, eficiente e qualitativa dos cuidados de saúde (Häyrynen et al., 2008).

Os tipos de sistemas de processos clínicos eletrónicos definidos pela *International Organization for Standardization* (2005) estão presentes na Tabela 2.

Tabela 2 - Tipos de sistemas PCE (Adaptado de International Organization For Standardization (2005))

Sistema PCE	Definição
<i>Electronic medical record</i> (Registo clínico eletrónico)	Geralmente focado nos cuidados médicos providenciados.
<i>Electronic patient record</i> (Registo de utente eletrónico)	Contém a maioria da informação clínica dos utentes de um determinado hospital.
<i>Computerized patient record</i> (Registo de utente informático)	Contém a maioria da informação clínica dos utentes de um determinado hospital.
<i>Electronic health care record</i> (Registo de cuidados clínicos eletrónico)	Contém toda a informação de saúde dos utentes.
<i>Personal health record</i> (Registo clínico pessoal)	Informação controlada pelo utente e contém informação, parcialmente ou totalmente, introduzida pelo próprio utente.
<i>Computerized medical record</i> (Registo clínico informático)	Gerado pelo <i>scan</i> de imagem de um processo clínico em papel.
<i>Digital medical record</i> (Registo clínico digital)	Um registo mantido na Web por um prestador de serviços de saúde.

Sistema PCE	Definição
<i>Clinical data repository</i> (Repositório de dados clínicos)	Uma base de dados que mantém e gere dados clínicos recolhidos de um prestador de serviços clínicos.
<i>Electronic client record</i> (Registo de cliente eletrónico)	O âmbito é definido por profissionais de saúde (ex: fisioterapeutas, assistentes sociais, etc.) em vez dos médicos.
<i>Virtual EHR</i> (Processo clínico eletrónico virtual)	N/A
<i>Population health record</i> (Registo clínico da população)	Contém dados agregados da população.

Os sistemas de processos clínicos eletrónicos podem ser usados em diferentes tipos de cuidados de saúde existentes num determinado país. Segundo a Entidade reguladora da saúde (2011), em Portugal existem quatro tipos de cuidados:

Cuidados primários: cuidados providenciados na comunidade por médicos de família. Primeira interação dos utentes com o Serviço Nacional de Saúde, logo esses cuidados são levados o mais proximamente possível dos locais onde as famílias ou indivíduos vivem.

Cuidados secundários: cuidados providenciados por um especialista, após reencaminhamento de um médico que presta cuidados primários de saúde.

Cuidados continuados: também conhecidos como cuidados terciários, dizem respeito ao acompanhamento de um doente por um conjunto de especialistas, de forma a acompanhar o tratamento receitado e centrado na recuperação do indivíduo.

Cuidados domiciliários: cuidados prestados a pessoas doentes ou inválidas em domicílios, lares ou instituições.

Os sistemas PCE podem ser utilizados por diferentes profissionais de saúde e *staff* administrativo. Entre os profissionais de saúde que podem usar os sistemas encontram-se os enfermeiros, médicos, radiologistas, farmacêuticos, técnicos de laboratório e radiologistas. Os utentes ou os seus pais (tutores), também podem utilizar os sistemas PCE (Häyrinen et al., 2008).

O armazenamento da informação relevante dos processos de negócio por parte das organizações de cuidados de saúde em formato eletrónico, deixa disponível um grande conjunto de dados, sobre os quais podem ser aplicados um conjunto de técnicas de *data mining* para encontrar padrões relacionados com eventos adversos, erros e custos desnecessários escondidos na estrutura dos processos clínicos. A análise dos resultados da aplicação dessas técnicas permite identificar as causas do mau desempenho de algum processo e proceder à sua otimização (Ruffolo et al., 2007).

Pagliari et al. (2007) referem um conjunto de benefícios potenciais, representados na Tabela 3, que advêm da utilização de sistemas PCE.

Tabela 3 - Benefícios potenciais sistemas PCE (Adaptado de Pagliari et al. (2007))

Tipo	Benefícios
<i>Participação dos Utentes (controlo, educação)</i>	<ul style="list-style-type: none"> • Maior responsabilidade na manutenção da saúde e cuidados pessoais; • Melhor seguimento dos tratamentos; • Maior participação na tomada de decisão; • Maior conhecimento de saúde adquirido.
<i>Melhorias na saúde</i>	<ul style="list-style-type: none"> • Melhor saúde e qualidade de vida.
<i>Qualidade dos cuidados</i>	<ul style="list-style-type: none"> • Melhoria dos relacionamentos com os profissionais de saúde (comunicação, confiança); • Acesso mais flexível a serviços (marcações de consultas e resultados de testes); • Aumento de segurança dos doentes (menos erros cometidos, resultados mais rápidos, alertas de medicamentos); • Melhor capacidade de seguir o tratamento dos utentes; • Melhorias na colaboração entre médicos e outros profissionais de saúde (enfermeiros, farmacêuticos, etc.); • Menos consultas desnecessárias, o que permite a redução das listas de espera; • Redução de custos (Redução do uso de serviços e tratamentos)

Analisando a Tabela 3, percebemos que os sistemas PCE têm potencial de melhorarem significativamente a qualidade dos serviços, clínicos e administrativos, prestados pelas instituições de saúde, facilitam a monitorização dos utentes e permitem que estes possuam mais controlo e responsabilidade nos tratamentos e cuidados associados.

Sendo um sistema informático de armazenamento de informação sensível e particular, um sistema PCE deve assegurar um conjunto de requisitos não funcionais que permitam aos utilizadores confiarem no sistema desenvolvido. Hoerbst & Ammenwerth (2010) referem como essenciais os seguintes requisitos não funcionais:

- **Transparência:** apresentar informação sobre o uso que será dado aos dados e quem os poderá visualizar;
- **Fiabilidade:** capacidade do sistema se manter funcional, mesmo que ocorram falhas;
- **Acuidade:** informação apresentada está de acordo com os dados inseridos por utentes ou profissionais de saúde; facilidade na atualização de informação;

- **Segurança:** garantia que os dados inseridos no sistema estão seguros e só podem ser visualizados por utilizadores autorizados;
- **Interoperabilidade:** capacidade do sistema em permitir que diferentes organizações possam aceder aos dados, caso lhes seja garantida autorização;
- **Eficiência:** capacidade do sistema em realizar as suas operações no menor tempo e com o menor gasto de recursos possível;
- **Usabilidade e acessibilidade:** uma significativa fatia dos utentes são indivíduos mais idosos com mais dificuldades de adaptação a novas tecnologias, logo é necessário apresentar uma *interface* agradável e fácil de aceder e utilizar;
- **Manutenção:** o sistema deve ser fácil de manter, pois um sistema deste tipo implica o armazenamento de um grande número de registos, o que pode levar a perdas de desempenho a médio prazo, caso certas precauções não sejam asseguradas;
- **Portabilidade:** o sistema deve ser fácil de utilizar em qualquer sistema operativo ou dispositivo.

Quando um utente divulga informação pessoal para ser armazenada em formato eletrónico, esse utente espera que a informação siga os princípios de confidencialidade médico-paciente. Logo, é necessário garantir que os sistemas informáticos possuam mecanismos que assegurem a confidencialidade, integridade e segurança dos dados partilhados pelos utentes, para que a população não perca a confiança nos sistemas eletrónicos usados pelas organizações de cuidados de saúde (Veiga, 2015).

3.4.1 Exemplo de aplicação – Centro Hospitalar do Porto

Segundo o Ministério da Saúde (2009), a falta de interligação de dados e de uma estratégia de Sistema de Informação, levou o Hospital Santo António a desenvolver uma estratégia de integração e interoperabilidade em parceria com a Universidade do Minho. Alguns dos objetivos do projeto desenvolvido foram a criação de um registo de dados estruturados, seguros e consistentes, registo de situações clínicas dos utentes e respetivas avaliações, fundamentação das decisões tomadas por parte dos profissionais de saúde, suporte de ações de prevenção e promoção de saúde e satisfação dos requisitos legais e profissionais.

O sistema desenvolvido possui três áreas, os alertas, lista de problemas ativos e o processo clínico eletrónico. A área do PCE inclui as notas clínicas, diários clínicos, procedimentos, intervenções

cirúrgicas, prescrições de tratamentos, exames e medicamentos, monitorização dos utentes e armazenamento e visualização de documentos (Ministério da Saúde, 2009).

A implementação desta solução no CHP trouxe vários benefícios para a organização, entre eles o aumento do rigor e fiabilidade dos dados registados, diminuição acentuada de erros e omissões, suporte à investigação clínica, migração dos processos do formato papel para o formato eletrónico, aumento da segurança do armazenamento da informação e acesso mais rápido à informação dos cidadãos, prescrições de medicamentos e tratamentos (Ministério da Saúde, 2009).

3.4.2 Exemplo de aplicação – Urgência Pediátrica Integrada do Porto

A Urgência Pediátrica Integrada do Porto foi projetada com o objetivo de dar uma resposta eficaz ao atendimento das crianças e dos adolescentes com doenças agudas na área do Porto, de forma a articular os vários níveis de cuidados existentes e promovendo a melhor assistência no local mais adequado (Ministério da Saúde, 2009). De maneira a facilitar a articulação, segundo o Ministério da Saúde (2009) o sistema de informação desenvolvido no âmbito da UPIP teve como principais requisitos:

- Criação de um repositório regional (rede de instituições), que possui os dados mais relevantes do histórico clínico dos utentes, como a lista de diagnósticos/problemas, medicação crónica, comentários clínicos e dados administrativos de cada utente;
- Permitir que os médicos possam aceder à informação do repositório a partir de qualquer uma das instituições da rede e a partir dos sistemas de informação utilizados nessas instituições;
- Implementar a referenciação do utente por via eletrónica para os diferentes níveis de rede;
- Definição das prioridades no sistema de triagem pediátrica na Urgência Pediátrica do Porto;
- Maximizar a articulação e integração do repositório de dados da rede UPIP e do respetivo SI, com os sistemas de informação em funcionamento nas instituições que integram a rede, de forma a que os diferentes processos de articulação e consulta de dados sejam transparentes para os profissionais de saúde;
- Permitir que o gestor do sistema possa monitorizar o funcionamento do sistema e obter um conjunto de informação através da definição de um conjunto de indicadores de desempenho;
- “Vender” o sistema aos profissionais de saúde, de forma a garantir que participam no processo, mostrando os benefícios na articulação e complementaridade da prestação de cuidados.

O funcionamento da rede UPIP baseia-se em três níveis, o primeiro nível integra todos os centros de saúde do concelho do Porto, Maia, Matosinhos, Gondomar e Valongo. O segundo nível engloba os

locais de Atendimento Pediátrico Referenciado a funcionarem no Centro Hospitalar do Porto e na Unidade Local de Saúde de Matosinhos. O terceiro nível integra a Urgência Pediátrica do Porto, a funcionar no Serviço de Urgência Pediátrica do Hospital de S. João (Ministério da Saúde, 2009).

O SI desenvolvido no âmbito da UPIP garantiu a melhoria da comunicação entre profissionais de saúde das instituições pertencentes à rede UPIP, através da partilha de informação clínica e administrativa acumulada nos diferentes contactos dos utentes com a rede. A melhoria na comunicação tem como benefícios, a diminuição dos tempos de espera, cuidados de proximidade, criação de um sentido de equipa dos profissionais abrangidos e desmaterialização de referenciação entre os diferentes níveis (Ministério da Saúde, 2009).

3.4.3 Integração de *Workflows* clínicos

As dificuldades em integrar os *workflows* clínicos em sistemas PCE tem contribuído para a sua baixa adoção por parte dos profissionais de saúde. Para ultrapassar os desafios com que se deparam, os profissionais de saúde utilizam soluções alternativas às intencionadas pelos programadores do sistema (Patterson, Gurses, Simmons, & Gibbons, 2014).

Segundo Patterson et al. (2014), alguns dos desafios na integração dos *workflows* clínicos incluem a necessidade de iniciar sessão em múltiplos sistemas, o número de cliques envolvidos nos processos, dificuldades em processar ordens fora do padrão habitual do sistema, apresentação da informação, como por exemplo a *interface* para o histórico de medicação de um doente, receitas e notas não ordenadas e falta de capacidade do sistema em permitir alterações nos dados clínicos.

Para Zheng et al. (2015), a integração de *workflows* clínicos em sistemas PCE influenciam a alocação de tempo entre tarefas por parte do *staff* clínico e administrativo, diminuem a necessidade de *multitasking* por parte do *staff* e a tendência dos clínicos em procurarem soluções alternativas para executarem os *workflows* e aumentam a eficiência dos *workflows*, principalmente nas atividades de recolha de informação dos utentes, comunicação com o *staff* relativamente a cuidados a ter nos tratamentos dos doentes e número de consultas por dia.

Para Bowens et al. (2010), existem um conjunto de elementos fundamentais para a integração com sucesso de *workflows* clínicos em sistemas PCE. Bowens et al. (2010) referem a importância da avaliação e desenho dos *workflows*, de forma a que os clínicos possam tomar medidas para diminuir interrupções ou produtividade na integração com o sistema PCE. A análise à avaliação dos *workflows* permite aos profissionais de saúde perceberem se é necessário procederem a mudanças nos fluxos de

trabalho para otimizar a produtividade dos mesmos após a integração. Para Bowens et al. (2010), a perspectiva da liderança da organização em relação à mudança é essencial, pois a integração de *workflows* clínicos num sistema PCE pode implicar alterações em elementos internos na organização e caso os gestores de topo não apoiem as mudanças necessárias, o resultado mais provável é o insucesso da integração. Outro aspeto fundamental é a promoção dos benefícios da integração para os profissionais de saúde e gestores de topo, como por exemplo, a melhoria na eficiência dos *workflows*, na segurança dos doentes e nos cuidados prestados. Por fim, Bowens et al. (2010) referem que os clínicos necessitam de treino adequado para que as tecnologias escolhidas possam ser integradas com o mínimo de constrangimentos para os *workflows* clínicos atuais.

3.4.4 *OpenEHR*

Segundo Buck et al. (2009), a utilização da abordagem *OpenEHR* permite a estruturação, gestão, armazenamento e comutação de dados dos utentes de uma forma segura e fiável entre diferentes organizações de saúde ou outros grupos de interesse. Esta abordagem tem como ideia base separar a definição de conceitos de *softwares* concretos ou Base de dados relacionadas com os processos clínicos eletrónicos e realocá-los em bibliotecas normalizadas. Essas bibliotecas são denominadas Arquétipos (Buck et al., 2009).

O *OpenEHR* utiliza uma modelação de dois níveis. Os Arquétipos são o *top level* e representam conceitos clínicos. A utilização de uma biblioteca independente e normalizada retira a necessidade de modelar os mesmos dados mais de uma vez e permite que profissionais de saúde que não possuam experiência tecnológica possam modelar os arquétipos dos seus sistemas PCE (Atalag, Yang, Tempero, & Warren, 2011). Para permitir a troca de informação entre sistemas com línguas diferentes, os Arquétipos são modelos independentes de linguagem (Buck et al., 2009). O *lower level* é o Modelo de Referência que consiste num conjunto de modelos técnicos que retratam as características genéricas dos registos de saúde, como por exemplo estruturas, tipos de dados e informação para que cumpram com os requisitos éticos e legais (Atalag et al., 2011).

Na Figura 4 é possível observar o esquema da metodologia de dois níveis utilizado pelo *OpenEHR*. No centro da imagem encontra-se o Sistema de Informação que guarda a informação e comunica-a para outros sistemas. No lado esquerdo estão representados os arquétipos, ou seja, as bibliotecas com os conceitos de conhecimento. O *software* e os modelos da base de dados do sistema

encontram-se no lado direito, assim como os modelos formais ou linguagens que são instâncias dos conceitos de conhecimento.

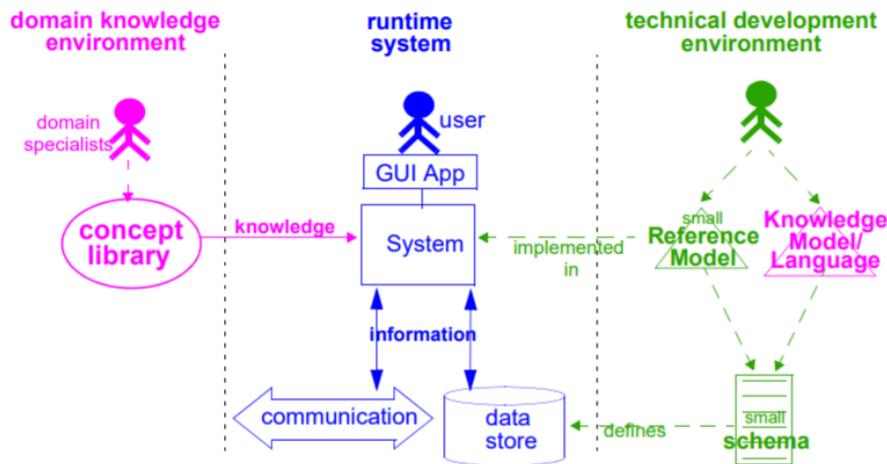


Figura 4 - Metodologia Dois-Níveis (retirado de Beale (2001))

Para Atalag et al. (2012), a utilização de *OpenEHR* é benéfica pois permite a interoperabilidade semântica com outros sistemas de informação de saúde, a estrutura para a captura de requisitos clínicos é intuitiva e permite que os profissionais de saúde possam participar no processo e a partilha de artefactos reutilizáveis. Atalag et al.(2012) também referem como vantagem o suporte às ferramentas de modelação, que, na maioria, são grátis e *open-source*.

Segundo Assis & Souza (2014), a utilização de *OpenEHR* nos sistemas de saúde oferece vantagens técnicas, na redução da necessidade de manutenção do sistema e ao nível da abstração de dados clínicos. O processo de comutação e acesso à informação entre diferentes organizações de saúde é mais rápido e menos burocrático. Assis & Souza (2014) referem algumas dificuldades encontradas, sendo a principal a transformação dos sistemas, pois no repositório de Arquétipos nem sempre é possível encontrar um que responda às necessidades do sistema, o que implica a criação de um Arquétipo novo juntamente com os profissionais de saúde.

3.5 Desenvolvimento do Artefacto

3.5.1 Requisitos

Para Mohammad et. al (2010), os requisitos de *software* estabelecem as expectativas das funcionalidades a desenvolver numa determinada aplicação. A análise de requisitos engloba as interações com outras aplicações e Base de dados, requisitos de desempenho e requisitos da *interface*

do utilizador. Segundo Wieggers & Beatty (2013), “*requisitos são uma especificação do que deverá ser implementado. Os requisitos são descrições de como o sistema se deverá comportar, ou de uma propriedade ou atributo do sistema. Eles podem ser uma limitação no processo de desenvolvimento do sistema.*”

Para Wieggers & Beatty. (2013), os requisitos de *software* possuem três níveis, requisitos de negócio, requisitos do utilizador e requisitos funcionais. Todos os sistemas também devem possuir um conjunto de requisitos não funcionais.

- **Requisito de negócio:** Um objetivo de negócio da organização para o desenvolvimento de um produto ou serviço para um determinado cliente;
- **Requisito do utilizador:** Um objetivo ou tarefa que os utilizadores devem conseguir desempenhar no sistema;
- **Requisitos funcionais:** Descrição de um comportamento que o sistema deve exibir sob condições específicas;
- **Requisitos não funcionais:** Descrição de uma propriedade ou característica que o sistema deve exibir ou uma limitação que deve respeitar.

Na Figura 5 é possível observar as diferentes fases do desenvolvimento e gestão de requisitos.

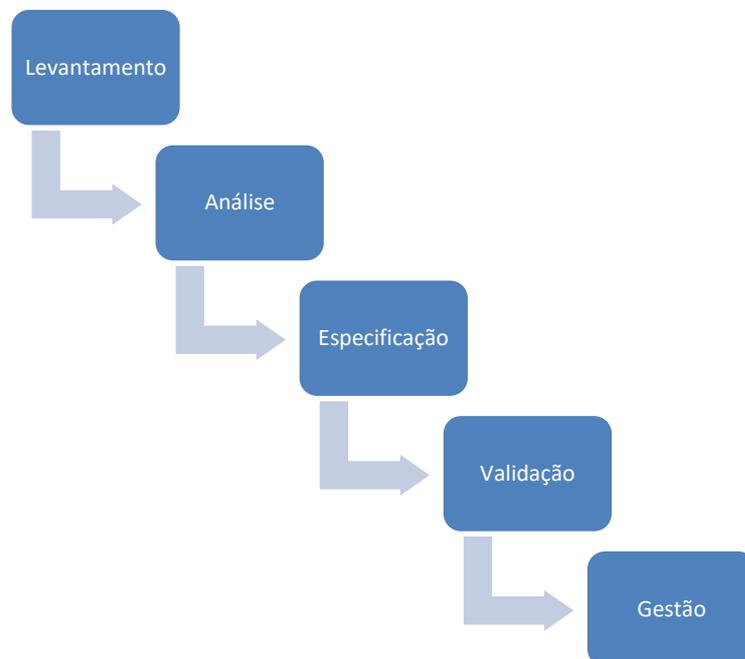


Figura 5 - Fases do Desenvolvimento e Gestão de requisitos (Adaptado de Wieggers & Beatty (2013))

As atividades chave no levantamento de requisitos são a identificação dos utilizadores e *stakeholders* do produto ou serviço, compreensão dos objetivos e tarefas dos utilizadores, como essas

tarefas se alinham com os objetivos de negócio, estudar o ambiente em que o produto ou serviço irá ser utilizado e trabalhar com os possíveis utilizadores do sistema para compreender as necessidades funcionais e expectativas de qualidade (Wiegers & Beatty, 2013).

A análise de requisitos envolve uma compreensão mais precisa de cada requisito. Para desenvolver essa compreensão é necessário analisar a informação recebida dos utilizadores para distinguir os objetivos das tarefas dos requisitos funcionais, expectativas de qualidade, regras de negócio e soluções sugeridas. Outras atividades da análise de requisitos incluem a definição de prioridades na implementação dos requisitos e identificação de lacunas nos requisitos ou requisitos desnecessários (Wiegers & Beatty, 2013).

Na especificação dos requisitos, os dados recolhidos são traduzidos para requisitos escritos e diagramas de forma a que a audiência interessada os possa compreender, rever e utilizar (Wiegers & Beatty, 2013).

A validação envolve a revisão dos documentos de requisitos elaborados para corrigir algum problema, antes da equipa de desenvolvimento começar a implementá-los. Desenvolvimento de testes e critérios para confirmar que o produto ou serviço baseado nos requisitos satisfazem as necessidades dos utilizadores e atingem os objetivos de negócio definidos (Wiegers & Beatty, 2013).

Segundo Wiegers & Beatty (2013), a gestão de requisitos tem como objetivo antecipar e acomodar as mudanças expectáveis de forma a minimizar o seu impacto no projeto. Algumas atividades desta fase incluem, a avaliação do impacto de mudanças propostas nos requisitos e incorporação das mudanças aceites de forma controlada, manter os planos do projeto a par com os requisitos, definição das relações e dependências entre requisitos e acompanhamento do estado e alterações dos requisitos ao longo do projeto.

3.5.2 Base de dados

Segundo Elmasri & Navathe (2015), a BD é uma coleção de dados relacionados entre si. Os dados são factos que podem ser registados e possuem um significado implícito. Connolly & Begg (2005) referem “*uma BD é um grande repositório de dados que pode ser usado simultaneamente por vários departamentos ou utilizadores*”. Connolly & Begg (2005) alegam que as BD deixaram de pertencer a um departamento e passaram a ser uma partilha de recursos corporativos.

Para Elmasri & Navathe (2015), as base de dados ou sistemas de BD são um componente essencial na sociedade moderna. A maioria das pessoas encontra várias atividades, todos os dias, que

envolvem interações com BD, como por exemplo, ida a um banco para depositar ou levantar dinheiro, reservar um quarto de hotel, comprar um bilhete de avião ou realizar uma compra *online*. Estas interações são exemplos de aplicações de base de dados tradicionais, em que a maioria dos dados armazenada e acedida é numérica ou texto. A proliferação das redes sociais como o *Facebook* ou o *Twitter* trouxe a necessidade de criar BD capazes de armazenar dados não tradicionais como *posts*, *tweets*, imagens e vídeos. Estes novos sistemas de base de dados são referidos como sistemas de armazenamento de *big data* ou sistemas *NoSQL* (Elmasri & Navathe, 2015).

De seguida apresentam-se algumas propriedades das BD (Elmasri & Navathe, 2015):

- Deve representar um aspeto do mundo real;
- É uma coleção de dados lógica e coerente. Um conjunto de dados aleatórios só pode ser considerado como base de dados se possuírem relacionamentos entre si;
- Uma BD é projetada, construída e preenchida com um propósito específico em mente. Foca-se num grupo de utilizadores e no que estão interessados.

Segundo Elmasri et Navathe (2015), uma BD pode ser criada e mantida à mão ou informaticamente. Uma base de dados informatizada pode ser criada e mantida por um conjunto de aplicações ou por um sistema de gestão de base de dados (DBMS). Um DBMS é um sistema informático que facilita os processos de definição, construção, manipulação e partilha de bases de dados entre vários utilizadores e processos. Para Elmasri & Navathe (2015), “*definir uma BD envolve especificar os tipos de dados, estruturas e limites dos dados para serem armazenados na base de dados. Construir a BD é o processo de guardar os dados em algum meio de armazenamento controlado pela DBMS. Manipular a BD inclui funções como consultas à BD para recolher informação específica e atualizar a BD. A Partilha permite que múltiplos utilizadores e aplicações possam aceder simultaneamente à BD*”. Connolly & Begg (2005) afirmam que o DBMS permite aos utilizadores inserirem, atualizarem, apagarem e recolherem dados da BD através de uma linguagem de manipulação de dados. A linguagem mais utilizada atualmente é a *Structured Query Language*, mais conhecida como SQL. Connolly & Begg (2005) também referem que o SQL é a linguagem formal para DBMSs relacionais.

Elmasri & Navathe (2015) ilustram as fases principais da criação de uma BD num diagrama que pode ser observado na Figura 6.

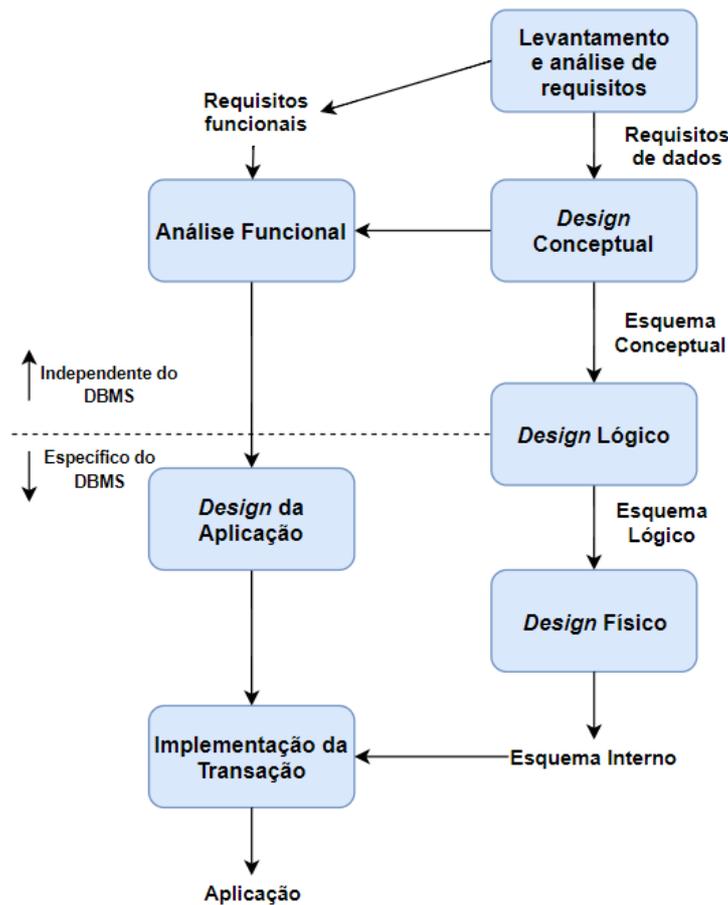


Figura 6 - Fases principais da criação de uma base de dados (Retirado de Elmasri & Navathe (2015))

O levantamento e análise de requisitos é essencial para o desenvolvimento da BD, mais concretamente para compreender e documentar as necessidades da informação que precisa de ser armazenada. Os requisitos funcionais da aplicação consistem nas transações que os utilizadores aplicam na BD, como a atualização, consulta ou inserção de dados (Elmasri & Navathe, 2015).

Após a recolha e análise dos requisitos é necessário criar o modelo conceptual da base de dados. Segundo Connolly & Begg (2005), esta fase descreve que dados são armazenados na BD, as relações entre eles, limitações e informação semântica sobre os dados e a segurança e integridade da informação. Elmasri & Navathe (2015) alegam que os modelos conceptuais facilitam a compreensão dos utilizadores sem experiência técnica, porque não incluem conceitos com detalhes da implementação. Elmasri & Navathe (2015) afirmam que o modelo conceptual deve ser usado como referência para garantir que todos os requisitos dos dados são garantidos e que não existem conflitos entre eles.

A fase lógica da construção da BD, também conhecida como mapeamento do modelo de dados, corresponde à implementação da base de dados utilizando um sistema de gestão de base de dados.

Grande parte dos DBMS existentes no mercado possuem a capacidade de transformar um modelo conceptual num modelo de dados lógico (Elmasri & Navathe, 2015).

Na última fase, o *design* físico, as estruturas internas de armazenamento, estruturas de ficheiros, indexamentos e caminhos de acesso para os ficheiros da BD são especificados. Em paralelo com estas atividades, são implementadas aplicações para fazer uso das transações da base de dados (Elmasri & Navathe, 2015).

Para Elmasri & Navathe (2015), o modelo relacional representa a BD como um conjunto de relações. Segundo Elmasri & Navathe (2015), “*cada relação assemelha-se a uma tabela de valores*”. Cada linha na tabela representa um conjunto de registos de dados enquanto o título e as colunas da tabela ajudam a interpretar o significado dessas linhas. Todos os valores numa coluna são do mesmo tipo. A terminologia formal para os modelos relacionais é a seguinte, a uma linha dá-se o nome de tuplo, o cabeçalho das colunas é um atributo, a tabela é uma relação e os tipos de valor que aparecem em cada coluna é representado por um domínio de valores possíveis. Connolly & Begg (2005) referem que o aparecimento do modelo relacional permitiu introduzir o conceito de relações normalizadas para lidar com problemas de semântica, consistência e redundância dos dados. Segundo Connolly & Begg (2005), a normalização “*é uma técnica para produzir um conjunto de relações com propriedades desejáveis, de acordo com os requisitos de dados de uma organização*”.

No contexto do *OpenEHR*, os arquétipos permitem a interoperabilidade semântica dos dados, ou seja, garantem a compatibilidade de terminologias, classificações e diretrizes clínicas informáticas entre diferentes sistemas, enquanto os modelos de referencia providenciam identificadores, tipos e estruturas de dados e vários padrões comuns de *design* que permitem a interoperabilidade na partilha de dados entre diferentes sistemas (Beale, 2001). Na construção da BD, os arquétipos utilizados devem ser respeitados no desenvolvimento do modelo conceptual, por exemplo, o nome de cada coluna deve respeitar as terminologias referentes aos arquétipos escolhidos. O desenvolvimento do modelo conceptual e lógico da BD deve ter em conta os modelos de referência do projeto, de forma a que os tipos e estruturas de dados criados estejam normalizados para facilitar a partilha de informação entre diferentes sistemas.

3.5.3 Aplicação *Web*

Para Taleb et al. (2010), inicialmente as aplicações *web* eram uma série de páginas de HTML estáticas e as interações entre os utilizadores e os servidores *web* eram muito limitadas. Atualmente, as

aplicações possuem serviços altamente interativos e a integração com outros *softwares* como bases de dados e repositórios online. Esta evolução foi possível devido a *scripts* do lado do servidor que tornaram as páginas *web* dinâmicas e permitiram mais interações entre os utilizadores e os vários sistemas informáticos (Huluka & Popov, 2012). Segundo Taleb et al. (2010), o *design* das aplicações organizacionais, atualmente, é muito influenciado pelos requisitos de negócio de cada uma das entidades (Taleb et al., 2010).

Taleb et al. (2010) referem que a *internet* oferece um conjunto de tecnologias e infraestruturas que permitem o desenvolvimento de aplicações interativas e flexíveis. Para Taleb et al. (2010) a população e organizações têm-se tornado cada vez mais dependentes das aplicações *web*, o que tem conduzido ao aumento da complexidade destes *softwares*.

Dingsøyr et al. (2012) referem que o desenvolvimento de *software* ágil baseia-se num conjunto de princípios, entre eles o foco principal na satisfação do cliente, aceitação de alterações tardias nos requisitos, períodos de entrega curtos de *software* funcional, trabalho conjunto entre o cliente e a equipa de desenvolvimento, desenvolvimento sustentável do *software*, análises e adaptações necessárias, por parte da equipa, para se tornar mais eficaz, atenção permanente à excelência técnica e um bom desenho da solução permitem aumentar a agilidade do desenvolvimento. Conboy (2009) define o desenvolvimento de *software* ágil como "*pronto para rapidamente e inerentemente criar mudança, adotar a mudança de forma pró-ativa ou reativa e aprender com a mudança e a sua contribuição na criação de valor para o cliente através dos seus componentes e relações com o ambiente.*"

No contexto desta investigação, a aplicação *Web* servirá como *interface* para os profissionais de saúde interagirem com o sistema, logo no desenvolvimento da aplicação deve ser atribuída importância aos aspetos de usabilidade e acessibilidade mencionados pelos clínicos nas reuniões, de forma a que a *interface* criada seja agradável e intuitiva. Segundo Schuler et al. (2006), a *interface* de um sistema de processo clínico eletrónico "*é, discutivelmente, a parte mais importante de um sistema PCE, pois é com ela que os profissionais de saúde interagem, sendo por isso de extrema importância para o seu sucesso*". A aplicação *Web* também deve possuir mecanismos de conexão à base de dados, para que os profissionais de saúde possam visualizar e interagir com a informação armazenada. Algumas dessas interações são a inserção, atualização e remoção dos dados.

3.6 Ferramentas de Modelação de Processos

Delgado et al. (2015) dividem a lista de características fundamentais de ferramentas de modelação de processos em dois módulos, técnico e não técnico. O módulo técnico contém um conjunto de requisitos do *software*, e Delgado et al. (2015) atribuem bastante importância ao “*suporte de padrões nos workflows, configuração da calendarização de todo o sistema, relacionar tarefas e documentos, arquitetura das ferramentas de modelação de processos, integração com redes sociais, trabalho colaborativo nos processos, lista de versões dos processos, monitorização de processos*”, entre outros. O módulo não técnico contém as características não relacionadas com o *software* e incluem “*pacotes de instalação, disponibilização de documentação, suporte a diferentes linguagens e atividade da comunidade*” (Delgado et al., 2015).

Segundo Delgado et al. (2015), as seguintes tecnologias de modelação de processos de negócio gratuitas são as mais utilizadas pelos modeladores.

- *Aris Platform*;
- *Bizagi*;
- *Bonita*.

Na Tabela 4 encontram-se algumas características das ferramentas listadas.

Tabela 4 - Características Ferramentas BPM (Adaptado de Bizagi (2014); Bonitasoft (2018), Community (2017))

Ferramenta	Características
<i>Aris Platform</i>	<ul style="list-style-type: none">• <i>Interface</i> intuitiva para o utilizador;• Modelos para o panorama de processos, processos de negócio, gráficos organizacionais, infraestruturas de TI e modelos de dados;• <i>Design</i> inteligente permite aos modeladores; capturarem a informação das empresas de forma fácil e rápida;• Modelo é criado imediatamente após a inserção dos dados necessário e pode ser alterado futuramente;• Possui fragmentos do modelo, que permite a reutilização de combinações de objetos na modelação;• Contém o conceito de “modelação sensível ao contexto”, que permite, com a ajuda de <i>mini toolbar</i> quais os símbolos que podem ser usados para conectar objetos.
<i>Bizagi</i>	<ul style="list-style-type: none">• <i>Drag and Drop interface</i> desenhada para qualquer utilizador possa desenhar processos facilmente;• Permite a importação e exportação para ferramentas de modelação de diagramas como o <i>Visio</i>.

Ferramenta	Características
	<ul style="list-style-type: none"> • Possibilidade de especificar diferentes aspetos técnicos, como por exemplo, a duração das transações, mensagens, notificações e chamadas externas ao sistema; • Permite a documentação dos processos e a publicação dos mesmos em formato <i>Word, PDF, SharePoint</i> e <i>Web</i>; • Encoraja o trabalho em equipa ao permitir que múltiplos utilizadores possuam um ambiente em que podem trabalhar em simultâneo, hospedado na <i>cloud</i> ou no próprio local de trabalho; • O <i>Bizagi</i> possui poderosas capacidades de simulação que permitem aos utilizadores tomarem melhores decisões após visualizarem o impacto das ideias e mudanças propostas; • Animações das simulações com <i>feedback</i> em tempo real, que ajudam os modeladores a identificar recursos sobre utilizados, elementos com recursos insuficientes e oportunidades para melhorias.
<i>Bonita</i>	<ul style="list-style-type: none"> • Permite o <i>design</i>, relacionar, executar, monitorizar e otimizar os processos de negócio; • <i>Drag and Drop interface</i>; • Integração com outras aplicações, como por exemplo <i>SQL Server</i> e <i>DBMSs</i>; • Ferramenta de modelação de dados do negócio; • Monitorização avançada; • Soluções para a produção de relatórios do estado dos processos de negócio

O estudo realizado por Delgado et al. (2015) focou-se na análise das características destes *softwares* e na comparação com as características apresentadas no início desta secção. Delgado et al. (2015) concluíram que, apesar de todas as ferramentas possuírem vantagens e desvantagens e a utilização de cada uma depender do contexto da aplicação, o *software Bonita* apresentou os melhores resultados sendo o mais competitivo e completo em relação às restantes ferramentas.

3.7 Soluções existentes de Gestão de *workflows* clínicos e *OpenEHR*

3.7.1 Gestão de *workflows* clínicos e tomada de decisão

Weidenhaupt & Lagor (2008) afirmam que os profissionais de saúde nem sempre chegam às mesmas conclusões ao analisar o mesmo conjunto de informação clínica. Os clínicos, por vezes, não

estão atualizados em relação à gestão de tratamentos a serem aplicados, especialmente se forem em áreas apenas parcialmente relacionadas com as suas especialidades. Para além disto, os enfermeiros ocasionalmente esquecem-se de passos de um tratamento ou realizam os passos de forma incorreta, como por exemplo, dar a medicação errada a um doente. Devido a estes problemas, os cuidados de saúde recebidos pelos utentes pode estar sujeito a várias variações e o número de erros clínicos que têm como causa a negligência são elevados (Weidenhaupt & Lagor, 2008).

Para Weidenhaupt & Lagor (2008), parte da solução para estes problemas passa pela normalização da gestão de *workflows* clínicos e a monitorização contínua da qualidade dos cuidados de saúde, de forma a reduzir o número de erros clínicos. Dito isto, Weidenhaupt & Lagor, (2008) decidiram criar uma solução desenhada para ajudar os clínicos a aderirem a uma prática de medicina normalizada e prevenirem erros clínicos através de sugestões aos profissionais de saúde.

Em relação à arquitetura do sistema, a solução liga-se à infraestrutura de Tecnologias de Informação dos cuidados de saúde e a outros sistemas, como por exemplo, um módulo de apoio à decisão, um sistema de gestão dos *workflows* e um sistema PCE (Weidenhaupt & Lagor, 2008). O sistema de apoio à decisão é configurado para calcular as probabilidades de um conjunto de diagnósticos diferentes baseados nas evidências demonstradas nos dados do doente e calcular sugestões de diagnósticos ou passos para um tratamento baseados no rácio benefício/risco. O sistema de gestão de *workflows* oferece uma representação informática das diretrizes clínicas a serem seguidas e mantém o estado atual das ações clínicas. O sistema PCE é configurado para permitir o acesso aos dados clínicos por parte dos profissionais de saúde e utentes (Weidenhaupt & Lagor, 2008). De acordo com Weidenhaupt & Lagor (2008), o sistema de gestão de *workflows* clínicos e apoio à tomada de decisão recebem os dados clínicos do sistema de informação do Hospital, combinando e reconciliando a informação clínica para uso na gestão do problema clínico de um utente e para a visualização através de uma *interface* gráfica dos dados.

Segundo (Weidenhaupt & Lagor, 2008), a *interface* gráfica da solução contém os seguintes aspetos:

- **Linha de visão geral:** Contém a informação mais importante dos doentes, como o nome, sexo, data de nascimento e localização atual;
- **Janela de descobertas clínicas:** Configurada para conter um registo de descobertas clínicas, eventos e ações pertinentes para um problema em particular. Estes registos são baseados na informação clínica;

- **Janela de diagnóstico diferenciado:** Permite a representação das probabilidades para um ou mais diagnósticos diferenciais, em função da informação clínica recolhida. Se novas informações forem inseridas, as probabilidades são atualizadas;
- **Janela de protocolo e mapeamento do tempo:** Providencia a representação gráfica de um *workflow* clínico, ajudando a perceber os protocolos a cumprir e o mapeamento temporal de cada ação a desempenhar. Este mapeamento inclui a correlação entre a progressão de um *workflow* clínico com o tempo decorrido desde que os sintomas de um doente surgiram e o tempo que resta, dentro de um certo período temporal, para o tratamento do problema clínico;
- **Janela de sugestões de próximos passos:** Tem como objetivo assistir o diagnóstico e decisões sobre os tratamentos a serem aplicados por um ou mais profissionais de saúde;
- **Janela explicatória:** Esta janela está configurada para oferecer uma explicação para um determinado “próximo passo” sugerido. A explicação pode incluir, uma descrição da razão para o “próximo passo” sugerido, derivado do módulo de apoio à tomada de decisão.

(Weidenhaupt & Lagor, 2008) afirmam que a solução apresentada suporta a gestão de doenças agudas num cenário complexo clínico. Isto é particularmente importante para doenças em que a aderência às diretrizes é crucial e são caracterizadas por decisões complexas durante o tratamento. Alguns exemplos destes casos são a gestão de pacientes com concussões, infarto do miocárdio e enfarte.

3.7.2 Solução baseada em arquétipos para monitorizar doentes com doenças crónicas

O uso de tecnologia de comunicação para monitorizar doentes e o seu estado de saúde é um ponto fulcral na melhoria da gestão de doenças crónicas (Rodríguez et al., 2014). Para Rodríguez et al. (2014), o uso da tecnologia tem como vantagens a redução de hospitalizações, melhoria dos cuidados próprios dos doentes e o armazenamento dos resultados de tratamentos podem ser valiosos para futuras pesquisas e análises.

De forma a detetar doenças no seu estado inicial, utentes com insuficiência cardíaca foram analisados, em casa e em condições ambulatoriais. Os utentes foram equipados com sensores com o objetivo de recolher múltiplos dados físicos como temperatura da pele, índice de transpiração, padrões de atividade e eletrocardiogramas. No total recolheram-se 67 parâmetros relevantes para doentes com insuficiência cardíaca (Rodríguez et al., 2014).

Segundo (Rodríguez et al., 2014), os arquétipos foram usados para modelar o conhecimento médico e providenciar interoperabilidade, normalização e computabilidade em relação aos processos

clínicos eletrônicos. Os arquétipos foram pesquisados de acordo com cada parâmetro médico necessário, sendo que alguns conceitos tiveram de ser adicionados, como por exemplo, a atividade diária e as mudanças de peso diárias, enquanto outros conceitos mais gerais, como testes de laboratório e cardiogramas, já constavam no repositório pesquisados. Os dados recolhidos foram armazenados em BD relacionais, utilizando a linguagem SQL (Rodríguez et al., 2014).

Devido à natureza dos arquétipos, foi necessário utilizar um componente para configurar os conceitos clínicos e os arquétipos correspondentes. Esse componente denomina-se configurador *OpenEHR* e simplifica as associações entre os parâmetros necessários por parte dos profissionais de saúde e os arquétipos armazenados no *OpenEHR kernel*. Este configurador desempenha um papel importante na automação do acesso aos processos clínicos eletrônicos, pois define as propriedades que permitem a comunicação correta entre a aplicação do lado do cliente e o *OpenEHR kernel* (Rodríguez et al., 2014).

Rodríguez et al. (2014) referem que a *interface* utilizada foi o *Slim MEST*, uma aplicação *web* que permite o acesso seguro aos profissionais de saúde, onde estes podem visualizar, analisar e atualizar os dados dos seus utentes. A aplicação acede ao processo clínico eletrônico, localizado no *OpenEHR kernel*, e permite a visualização dos dados recolhidos no período de observação. Esta aplicação depende dos arquétipos incluídos no PCE para se tornar mais robusta e flexível às variações introduzidas no *OpenEHR kernel*, para além de facilitar a adaptação a outro sistema *OpenEHR* em qualquer contexto (Rodríguez et al., 2014).

Rodríguez et al. (2014) concluem que o modelo *OpenEHR* oferece varios benefícios, como a normalização do PCE e a criação de uma sinergia efetiva com ferramentas que facilitam a participação e intervenção dos profissionais de saúde nos cuidados de pacientes com doenças crónicas, usando os arquétipos como base do sistema. Esta abordagem é especialmente significativa pois permite que os clínicos e doentes obtenham benefícios do progresso do uso da tecnologia de comunicação, devido às suas dinâmicas e simplicidade.

3.7.3 Integração de processos clínicos flexíveis utilizando a notação BPMN 2.0

Yao & Kumar (2013) referem que os *workflows* têm um papel importante no ambiente clínico, pois permitem delinear os passos do tratamento de um paciente. O facto dos *workflows* serem muito dinâmicos, dependentes do contexto, guiados por eventos e necessitarem de conhecimentos intensivos

torna-os bastante únicos. De forma a que o resultado final seja ótimo, é necessário manter a coordenação e o fluxo de informação entre as várias entidades que constituem o *workflow*.

A necessidade de encontrar novas técnicas de modelação para desenhar *workflows* flexíveis foi motivada por algumas considerações, como o esforço que tem sido realizado de forma a estabelecer a interoperabilidade nos cuidados de saúde, a representação de diretrizes partilháveis para a prática clínica e para modelos executáveis e flexíveis dos *workflows* clínicos. A execução destes é altamente dependente do conhecimento médico existente, recursos disponíveis e dados específicos do caso em questão, ou seja, diferentes profissionais de saúde com diferentes níveis de conhecimento e áreas de atuação podem oferecer diferentes tratamentos ao mesmo doente. Daí, a importância de formalizar a modelação destes cenários (Yao & Kumar, 2013).

O objetivo do trabalho desenvolvido por Yao & Kumar (2013) passou por demonstrar como se podem desenhar processos clínicos flexíveis, tendo formalizado o conhecimento médico em regras e, também a contextualização dos detalhes da informação, de forma a que os *workflows* clínicos que envolvem múltiplos participantes possam melhorar a qualidade dos cuidados de saúde.

A forma encontrada pelos autores Yao & Kumar (2013) para comprovar o objetivo da sua investigação passou pela criação de um Protótipo que permitisse o desenho e execução de *workflows* clínicos, de forma a comprovar a sua flexibilidade na atuação clínica. Com isso em mente, Yao & Kumar (2013) escolherem uma ferramenta BPM que possibilitasse o desenho e execução dos *workflows*, tendo selecionado a ferramenta *open-source Drools-Flow 5.2*. No desenho dos *workflows* foi utilizada a notação BPMN 2.0 por ser um modelo de referência na normalização da representação gráfica dos elementos do processo (Yao & Kumar, 2013).

Segundo Yao & Kumar (2013), a abordagem utilizada na investigação aqui descrita diferencia-se das restantes, pois é utilizada a linguagem de modelação de processos BPMN 2.0 para modelar as atividades clínicas e coordenar as interações entre as várias entidades dos cuidados de saúde, as execuções dos *workflows* são ativadas pelo contexto clínico, derivado de tarefas médicas anteriores e o ambiente atual. Esta informação contextual apenas está disponível durante a execução do *workflow* e é única para cada utente. Outro fator de diferenciação está relacionado com o apoio à decisão providenciado durante os encontros com os doentes é baseado no contexto formal e ontologias médicas alinhadas com as diretrizes clínicas, de forma a assegurar as recomendações médicas corretas.

Para Yao & Kumar (2013), a investigação que efetuaram permitiu demonstrar que existem novas abordagens para o *design* de sistemas de apoio à decisão, através da flexibilidade e adaptabilidade dos *workflows* clínicos, desde que estes tenham em conta o conhecimento médico na forma de regras e a

contextualização da informação. A investigação proposta promete vários tipos de benefícios para a comunidade médica, permitindo reduzir a incidência de erros de tratamento, o que se traduz na melhoria da segurança dos utentes. Por fim, podem ser geradas melhores e mais rápidas recomendações em vários pontos de decisão (Yao & Kumar, 2013).

CAPÍTULO 4. Protótipo da aplicação de gestão de *workflows* clínicos

De forma a cumprir com os objetivos da dissertação, dando especial ênfase à possibilidade de integração de uma ferramenta de *Business Process Management* no sistema de Processo Clínico Eletrónico do Centro Hospitalar do Porto, de forma a perceber se o modelo *OpenEHR* pode ser aplicado na configuração e gestão dos *workflows* clínicos, ficou definida a necessidade de desenvolvimento de um Protótipo que pudesse testar essa possibilidade e perceber de que forma se pode facilitar a interoperabilidade e flexibilidade dos *workflows* clínicos.

Numa fase inicial deste capítulo será possível observar as diferentes fases da construção do Protótipo e as atividades incluídas em cada uma dessas fases. De seguida, encontra-se uma descrição mais detalhada das atividades que pertencem à primeira fase da construção e termina com a visualização da arquitetura construída para o Protótipo, que facilita a compreensão da aplicação, permite visualizar as linguagens de programação que constituem a plataforma e como serão aplicadas e as ligações entre as diferentes ferramentas que constituem o Protótipo. Por fim, apresentam-se as atividades do desenvolvimento do Protótipo, que incluem diagramas de sequência para explicar as interações da aplicação, as *interfaces* disponíveis para os utilizadores e a lista de páginas criadas que contêm a lógica do Protótipo desenvolvido.

4.1 Fases da construção do Protótipo

Após uma análise do trabalho a efetuar na construção do Protótipo, que inclui a definição de requisitos, ferramentas e estrutura lógica da integração das diferentes ferramentas, a elaboração do Protótipo foi estruturada em duas fases, Conceção e Desenvolvimento, sendo que na Figura 7 é possível observar a representação destas fases.

A fase da Conceção engloba um conjunto de atividades de preparação para o desenvolvimento do Protótipo. Inicialmente, foi elaborada a recolha e análise de dados e requisitos sobre a solução a desenvolver, de forma a compreender qual a ferramenta BPM mais adequada para o projeto. Esta fase também inclui as linguagens de programação utilizadas no decorrer do desenvolvimento do Protótipo e a Arquitetura que detalha as ligações entre as diferentes ferramentas.

O Desenvolvimento do Protótipo contém as configurações das ferramentas utilizadas neste Protótipo, assim como a lógica inerente a essas definições. Após a escolha das ferramentas a utilizar no desenvolvimento do Protótipo, foi criada uma Base de dados que pudesse simular a existente no sistema

PCE e a comunicação que deve existir com a ferramenta BPM e aplicação *web*. A definição da ferramenta BPM permitiu a criação de *workflows* clínicos para ir testando a comunicação com a aplicação de gestão de *workflows* clínicos e possibilitou que a solução fosse sendo validada com os elementos de contacto no CHP e orientadores do projeto de dissertação, de forma a perceber se o caminho a ser seguido está de acordo com os requisitos estabelecidos e com as necessidades do hospital.

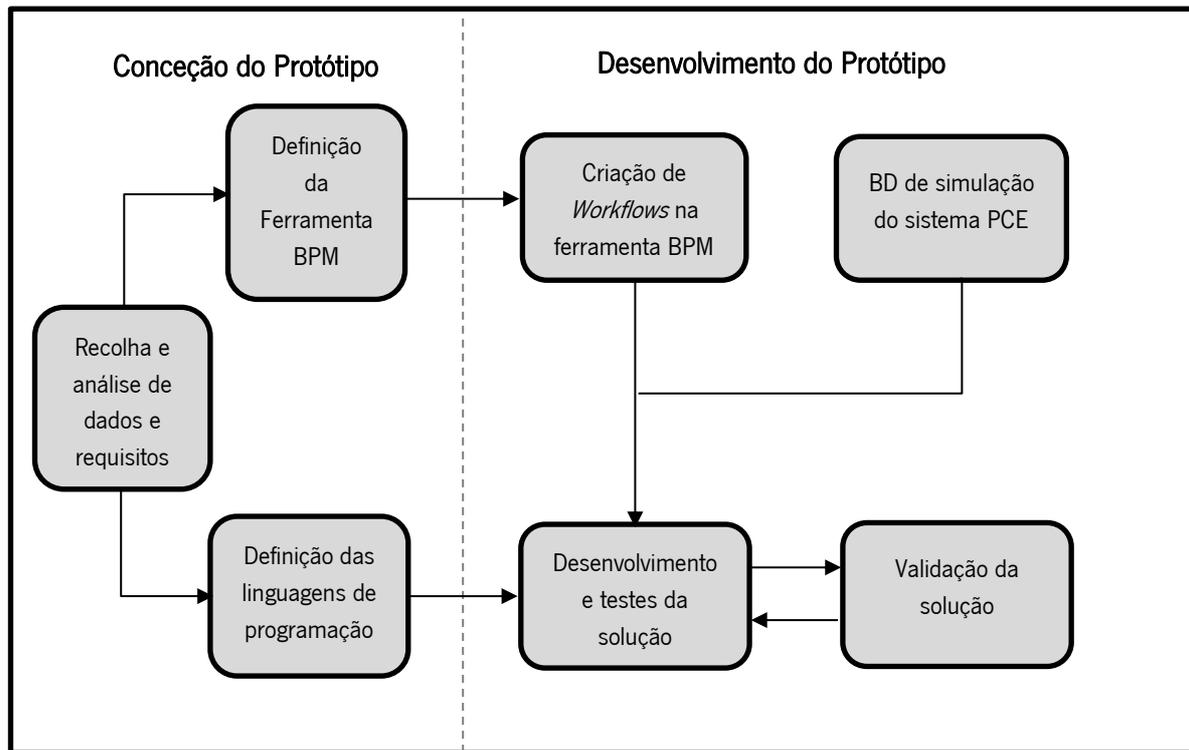


Figura 7 - Fases da construção do Protótipo

4.2 Conceção do Protótipo

Nesta secção, o objetivo passa por demonstrar os passos seguidos na preparação do desenvolvimento do Protótipo. Numa fase inicial foi necessário definir os requisitos do Protótipo com a equipa de desenvolvimento do CHP, que está a trabalhar no sistema de Processo Clínico Eletrónico. Esta primeira fase permitiu listar um conjunto de ferramentas essenciais durante o desenvolvimento, como a ferramenta BPM que contém as regras de negócio dos *workflows* e as linguagens de programação que possuem as regras sintáticas e semânticas da aplicação desenvolvida. Entretanto, foi realizada uma pequena análise aos requisitos da máquina local, de forma a perceber se suporta os produtos apresentados na lista de ferramentas e, por fim, a representação gráfica da Arquitetura, elaborada com o objetivo de facilitar o entendimento das interações do Protótipo desenvolvido.

4.2.1 Definição dos requisitos do Protótipo

O Protótipo construído no âmbito desta dissertação teve como objetivo o desenvolvimento de uma solução que permitisse a inclusão dos *workflows* clínicos no sistema PCE presente no Centro Hospitalar do Porto. Depois de construído, o Protótipo foi avaliado, de forma a analisar a sua utilidade no âmbito do centro hospitalar.

Os requisitos do Protótipo foram propostos por elementos pertencentes à equipa de desenvolvimento do sistema PCE, de forma a complementar o sistema com uma solução que permitisse a integração de *workflows* clínicos. Para a construção do Protótipo, definiram-se os seguintes requisitos:

- Utilização de uma ferramenta, que permita a configuração de *workflows* clínicos;
- A ferramenta deve ser gratuita;
- A ferramenta escolhida deve permitir que os *workflows* possam ser executados em aplicações externas, de forma a assegurar a interoperabilidade do Protótipo;
- A ferramenta deve permitir a integração de base de dados externas à própria ferramenta;
- O Protótipo deverá permitir a integração de formulários externos, baseados no módulo *OpenEHR*;
- O Protótipo deverá ser flexível, ou seja, refletir as mudanças nos *workflows* que vão sendo efetuadas na ferramenta;
- O Protótipo deverá possuir um controlo de utilizadores no acesso à aplicação *web*;
- O Protótipo deve permitir ao utilizador visualizar a informação sobre os *workflows* em que trabalhou;

Durante a construção do Protótipo, a equipa de desenvolvimento do sistema PCE foi avaliando a aplicação *web*, de forma a validar se o autor deste documento ia cumprindo com os requisitos delineados.

4.2.2 Lista de Ferramentas

Este ponto do relatório divide-se em três partes. Numa fase inicial é apresentada a ferramenta BPM escolhida após comparação feita entre as diferentes tecnologias de modelação de processos de negócio. Esta plataforma foi essencial no desenvolvimento do Protótipo pois contém os *workflows* de controlo de formulários. De seguida apresentámos a lista de linguagens de programação selecionadas para o desenvolvimento do Protótipo. Estas linguagens permitiram o desenvolvimento da *interface* para os utilizadores e a comunicação com a ferramenta BPM designada. Por fim, são apresentadas as

ferramentas escolhidas para a gestão da Base de dados e ambiente de desenvolvimento do aplicação *web* que compõem a fase de Desenvolvimento do Protótipo.

Ferramenta BPM

Na definição de requisitos ficou definido que a ferramenta escolhida não poderia acarretar custos, o que eliminou várias escolhas possíveis e deixou um leque constituído, na grande maioria, por versões *community* das ferramentas, que limitam algumas das funcionalidades do produto. As ferramentas analisadas mais detalhadamente neste processo foram destacadas na secção 3.6 e um produto recomendado pelos orientadores denominado *ProcessMaker*.

Após uma análise dos prós e contras dos quatro produtos, a escolha acabou por recair no *ProcessMaker*, escolha essa justificada pelas funcionalidades disponibilizadas por esta ferramenta na sua versão *community* que não colocaram entraves à implementação dos requisitos definidos para o Protótipo, como por exemplo, a ausência de limites na criação de processos, tarefas e utilizadores do sistema. Mas a principal razão para esta escolha deveu-se à análise feita à API e documentação disponibilizada pelos diferentes produtos, sendo que nestes aspetos o *ProcessMaker* ganhou vários pontos devido à informação mais vasta e acessível para os utilizadores sobre o funcionamento da plataforma e respetiva API, facilitando o entendimento e manuseamento da plataforma.

A representação dos *workflows* clínicos foi elaborada nesta ferramenta. Este produto disponibiliza uma API REST, construída em PHP, que permite o acesso remoto ao *ProcessMaker* através de *scripts* externos, para além de um alargado número de *endpoints* que permite que a maioria das ações possam ser realizadas tanto na *interface* do *ProcessMaker* como remotamente. Outra funcionalidade interessante da plataforma é permitir a utilização de Base de dados externas à aplicação, sendo um importante meio de ligação entre o *ProcessMaker* e a BD do sistema PCE.

De forma a oferecer algum contexto ao modo de utilização da plataforma, referir que na criação de *workflows*, os processos possuem tarefas que podem ser associadas a utilizadores. Para iniciar uma instância de um processo, designada por Caso, apenas os utilizadores aos quais está atribuída a tarefa inicial desse processo podem dar início a essa instância.

A ferramenta necessita dos seguintes requisitos para ser instalada:

- *Windows 7, Windows 8, Windows 10, Windows Server 2012 R2;*
- *PHP 5.3, 5.4, 5.5, 5.6;*
- *MySQL 5.1, 5.5;*

- *Apache 2.2, 2.4.*

Na Tabela 5 é possível observar as versões instaladas de cada *software* presente nos requisitos da ferramenta BPM.

Tabela 5 - Versões instaladas

<i>Software</i>	Versão instalada
<i>ProcessMaker</i>	<i>3.2.1-community</i>
Sistema Operativo	<i>Windows 10</i>
PHP	5.6.31
<i>MySQL</i>	5.1.11
<i>Apache</i>	2.4

Linguagens de Programação

Na seleção das linguagens de programação a utilizar no desenvolvimento do Protótipo, convém referir que um dos aspetos principais na escolha da linguagem responsável pelo *Server-Side* e consumo da API REST do *Processmaker*, deveu-se à quantidade de documentação disponibilizada pela própria plataforma e à quantidade de informação nos fóruns oficiais do produto. Durante o levantamento dos requisitos, um dos elementos da equipa de desenvolvimento do sistema PCE referiu que existia liberdade na escolha das linguagens de programação, apesar de demonstrar alguma preferência pela utilização de algumas bibliotecas *Javascript* como o *JQuery*, *AJAX* e *Node JS* na comunicação com a ferramenta BPM devido a questões de integração. A escolha acabou por recair no PHP, pela questão já explicada anteriormente em relação à documentação disponibilizada pelo *ProcessMaker* e também devido a um maior conhecimento do autor deste Protótipo nessa linguagem e, como existe alguma limitação de tempo no desenvolvimento deste Protótipo, a escolha recaiu na opção mais segura. Dito isto, é importante referir que a equipa de desenvolvimento do sistema PCE não levantou qualquer problema em relação a esta seleção. Na Tabela 6 é possível observar a lista de linguagens utilizadas assim como o objetivo de cada uma delas.

Tabela 6 - Linguagens de Programação do Protótipo

Linguagem	Função
HTML, CSS, Javascript	Estas linguagens permitiram desenvolver o código executado do lado do Cliente, ou seja, foram utilizadas para criar a <i>interface</i> do Protótipo, permitindo a interação com os utilizadores.
PHP	O PHP é responsável por consumir a API REST do <i>ProcessMaker</i> , funcionando como elo de ligação entre a aplicação a desenvolver e a ferramenta BPM. Também funciona como meio de ligação com a BD do sistema PCE.
SQL	Linguagem padrão utilizada nas bases de dados. Permite realizar a consulta e modificação dos dados pertencentes à BD. No caso deste projeto, é utilizado na BD de simulação do sistema do CHP e no <i>ProcessMaker</i> caso seja necessário proceder a modificações que a API não cubra.

Outras Ferramentas

Para além das ferramentas já listadas nos pontos anteriores, também foi necessário selecionar uma tecnologia que permitisse gerir a Base de dados de simulação do sistema PCE e um ambiente de desenvolvimento para a aplicação *web*.

O *MySQL Workbench* foi a ferramenta utilizada para gerir a BD de simulação do sistema PCE. Esta tecnologia permite a criação e edição da BD, tabelas e campos. A ferramenta também permite inserir e editar dados diretamente na BD, caso seja necessário. A utilização deste produto deveu-se ao conhecimento, do autor desta Dissertação, da ferramenta e perceção que seria suficiente para os requisitos e necessidades do Protótipo.

A tecnologia utilizada no ambiente de desenvolvimento da aplicação *web* foi o *PhpStorm*. Esta tecnologia permitiu desenvolver o código necessário para o funcionamento do Protótipo, sendo especializada no desenvolvimento de programação *web*. Assim como a ferramenta de gestão da BD, a escolha desta ferramenta deveu-se a um conhecimento anterior do autor no manuseamento do produto e a sua utilidade no desenvolvimento de aplicações *web*.

Os requisitos das ferramentas mencionadas acima podem ser observados na Tabela 7.

Tabela 7 - Requisitos de ferramentas

Ferramenta	Requisitos
<i>MySQL Workbench</i>	<ul style="list-style-type: none"> • <i>Microsoft .NET Framework 4.5.2;</i> • <i>Visual C++ 2015 Redistributable for Visual Studio 2015;</i> • Disponível para Windows, Mac OS e Linux.
<i>PhpStorm</i>	<ul style="list-style-type: none"> • 1 Gb RAM Mínimo, 2 Gb RAM Recomendado; • 1024x768 resolução de ecrã mínima; • Disponível para Windows, Mac OS e Linux.

4.2.3 Características da infraestrutura disponível

Com o intuito de perceber se a máquina onde foi desenvolvido o Protótipo possuiria os requisitos mínimos para a utilização das ferramentas necessárias, foram levantadas as especificações da máquina, sendo que a análise elaborada pode ser observada na Tabela 8.

Tabela 8 - Especificações da Máquina

Especificações da Máquina	
Sistema Operativo	<i>Windows 10 Home</i> Versão 1803
Processador	<i>Intel(R) Core(TM) i7-6700HQ CPU @ 260GHz</i>
Memória RAM	16 Gb RAM
Monitor	Monitor Integrado (Resolução: 1920x1080)
Disco	SSD: 256Gb e HDD 1Tb
Placa Gráfica	<i>NVIDIA GeForce GTX 950M</i>

Após a comparação com os requisitos das ferramentas a serem utilizadas no desenvolvimento do Protótipo, chegou-se à conclusão que a máquina suporta a instalação e utilização de todas elas.

4.2.4 Arquitetura do Protótipo

Após a definição dos requisitos do Protótipo, a seleção da ferramenta BPM e linguagens de programação foi possível desenhar a arquitetura que funcionou como base para o desenvolvimento do Protótipo. Na Figura 8 pode ser visualizada a representação gráfica da Arquitetura, assim como as tecnologias a serem utilizadas no Protótipo.

O utilizador, no contexto deste Protótipo corresponde a um Profissional de Saúde ou Utente, é responsável por iniciar o *workflow* clínico. Para iniciar o processo, o utilizador escolhe o *workflow* que pretende executar e essa decisão é encaminhada para o *BackEnd* que trata de elaborar o pedido, através da API, ao *ProcessMaker* com o objetivo de criar uma instância do *workflow* selecionado e receber o formulário inicial associado a esse *workflow*. O *BackEnd* trata a informação recebida do *ProcessMaker*, em formato *JSON*, e interage com o *FrontEnd* de forma a apresentar o formulário ao utilizador. Esta interação é recorrente, até as decisões tomadas pelo utilizador, através dos *inputs* inseridos nos

formulários, levarem ao fim do *workflow* ou caso a tarefa atual do processo esteja designada a outro utilizador. Se a última situação referenciada ocorrer, então esse utilizador deverá continuar o *workflow*.

Em certos cenários, a relevância dos dados inseridos pelos utilizadores leva à necessidade da inserção de informação na BD de simulação do sistema PCE. Nesses casos, o *BackEnd*, para além de interagir com o *ProcessMaker*, também irá comunicar com a BD, elaborando um pedido à estrutura de armazenamento e recebendo uma resposta com o estado do pedido efetuado. Por vezes, é necessário que a ferramenta BPM comunique diretamente com a BD, de forma a recolher informação essencial para a continuação da execução dos *workflows*. Nesses casos a ferramenta faz o pedido através de um *trigger* e trata a resposta de acordo com o que o *workflow* necessita.

Falta ainda referir como foi tratado um dos requisitos principais definidos, a flexibilidade do Protótipo. Um dos principais objetivos da dissertação passou por garantir que os *workflows* criados na ferramenta BPM pudessem ser modificados e que essas alterações fossem refletidas no Protótipo desenvolvido. Este objetivo surgiu de forma a que as modificações nos *workflows* fossem mais acessíveis e rápidas, pois caso contrário, qualquer alteração implicaria mudanças no código desenvolvido, o que não seria uma solução prática a médio e longo prazo. Por essa razão, a utilização de uma ferramenta BPM facilita a flexibilidade dos *workflows* clínicos, pois facilita as suas alterações e como as execuções dos Casos no Protótipo baseiam-se nas chamadas à API do *ProcessMaker*, significa que a versão que está a ser executada é a mais recente do *workflow*.

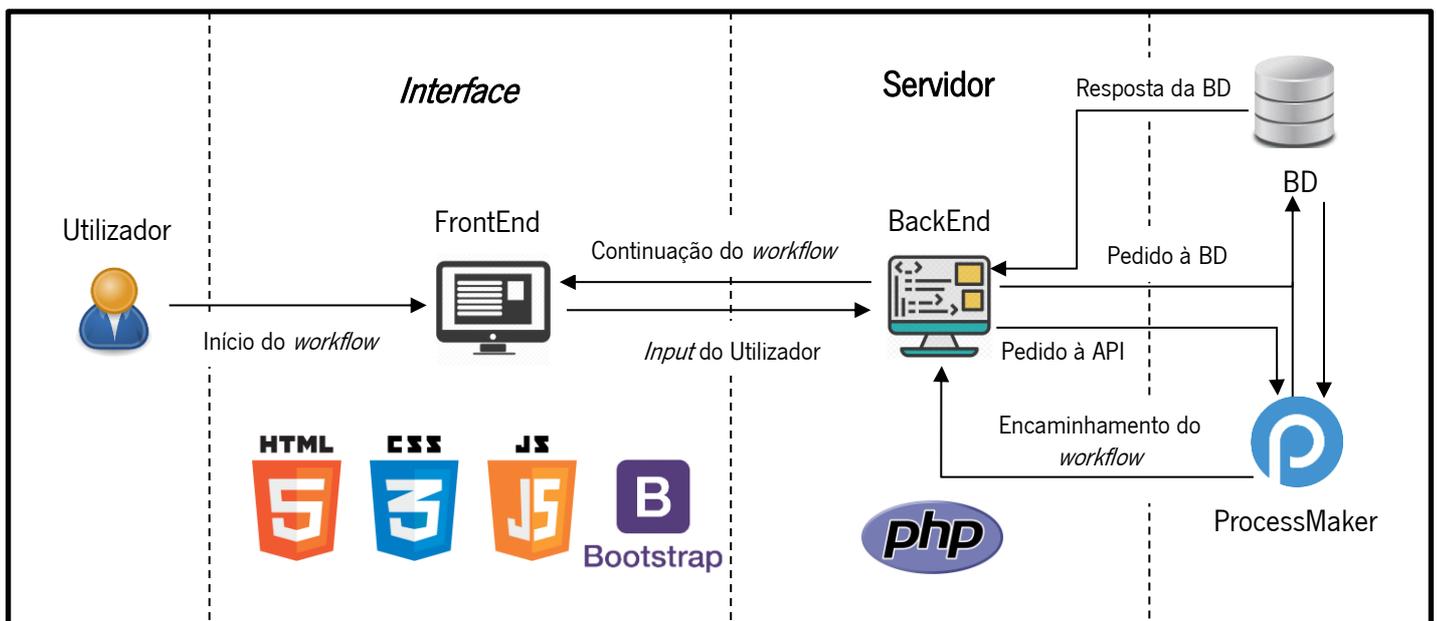


Figura 8 - Arquitetura do Protótipo

4.3 Desenvolvimento do Protótipo

Nesta fase da construção do Protótipo, o objetivo passou por desenvolver a aplicação *web* de gestão de *workflows* clínicos. Antes de passar para o desenvolvimento da aplicação referida, foi necessário criar as bases que suportassem a solução, sendo elas a Base de dados de simulação da que é utilizada no sistema Processo Clínico Eletrónico e a configuração do *ProcessMaker*, de forma a melhorar o entendimento do funcionamento da plataforma e como a usar mais eficientemente de maneira a cumprir os requisitos do Protótipo.

Esta secção começa com a informação relativa à criação da BD de simulação do sistema PCE, como o diagrama entidade-relacionamento e a descrição das tabelas e campos criados. De seguida, encontra-se a descrição das configurações realizadas no *ProcessMaker*, configurações essas relacionadas com a administração da ferramenta e do *design* de Processos. Por fim, será descrito o processo de desenvolvimento da aplicação de gestão de *workflows* clínicos, através da explicação detalhada da lógica utilizada na construção da aplicação *web* e na ligação da mesma à ferramenta BPM.

4.3.1 Base de dados

De forma a demonstrar as potencialidades do Protótipo, foi necessário criar uma Base de dados que simulasse a presente no sistema Processo Clínico Eletrónico do Centro Hospitalar do Porto. Esta BD foi imprescindível para o Protótipo desenvolvido, pois, em casos reais, a maioria dos *workflows* clínicos necessitam de consultar e manusear a informação presente no sistema PCE, de forma a que sejam executados corretamente. De referir que a BD foi criada tendo por base o modelo relacional e apenas contém as tabelas necessárias para a execução do *workflow* clínico fornecido pelos elementos do CHP, não sendo um exemplo completo da existente no sistema PCE.

Numa fase inicial, será apresentado o diagrama de entidade-relacionamento da BD, de maneira a demonstrar as relações entre as entidades criadas. De seguida, apresenta-se uma descrição das tabelas e campos que preenchem a BD de simulação do sistema PCE.

4.3.1.1 Diagrama de entidade-relacionamento

O DER é um fluxograma que pretende ilustrar como as entidades se relacionam entre si num sistema. Na Figura 9 é possível visualizar o DER criado para a BD de simulação do sistema PCE.

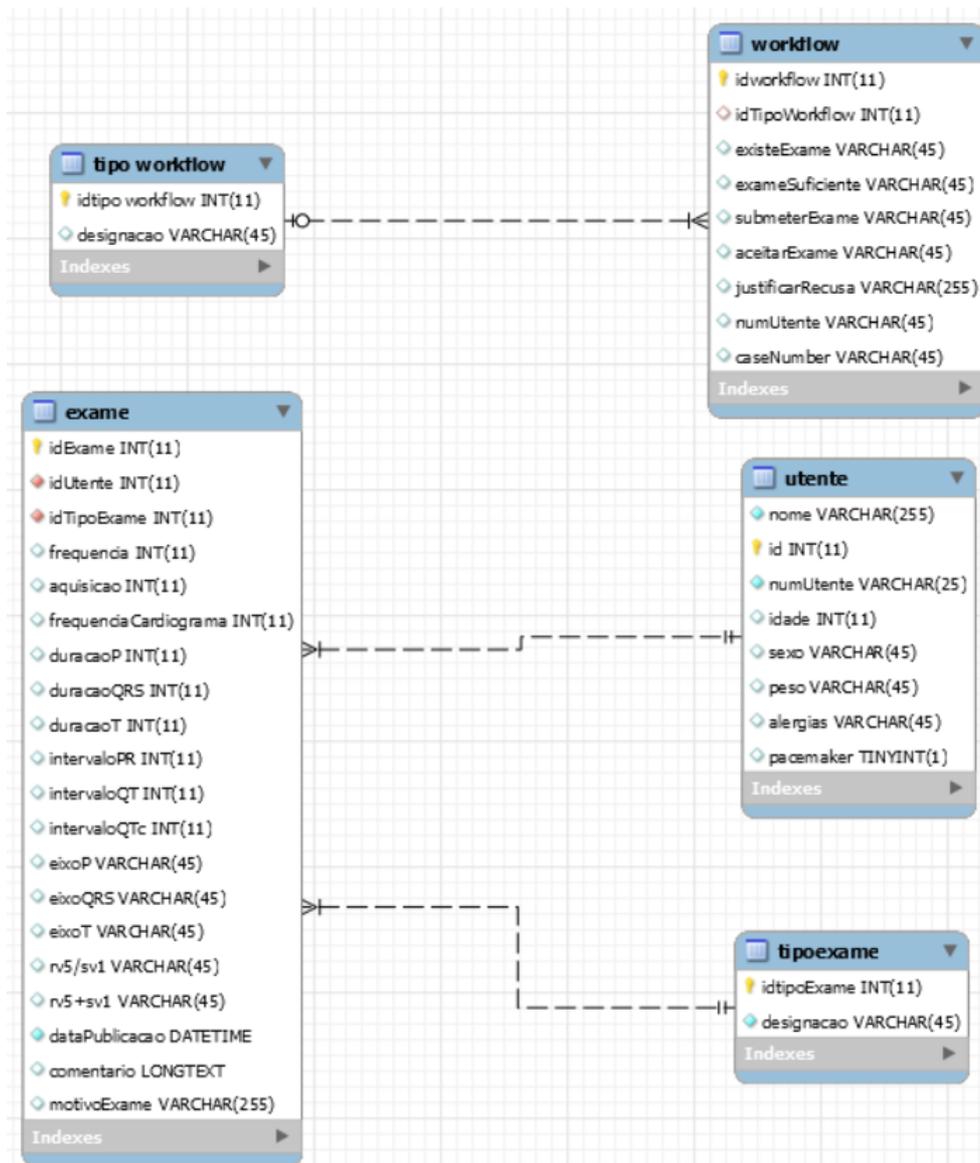


Figura 9 - Diagrama Entidade-Relacionamento BD

4.3.1.2 Descrição de entidades

Nesta subsecção, será realizada a descrição das entidades criadas na BD, contendo também o conjunto de campos que povoam essas entidades.

Tipo workflow

Contém os tipos de *workflow* presentes na ferramenta BPM *ProcessMaker*.

Tabela 9 - Entidade Tipo workflow

Atributo	Tipo de Dados	Limite	Característica	Observações	Exemplo
idTipoWorkflow	Int	11	Chave Primária	Não pode ser nulo; Auto incremento	1

Atributo	Tipo de Dados	Limite	Característica	Observações	Exemplo
designacao	Varchar	45	-	Não pode ser nulo	“Processo Exame TAC”

Workflow

Permite guardar os *workflows* executados no *ProcessMaker*.

Tabela 10 - Entidade Workflow

Atributo	Tipo de Dados	Limite	Característica	Observações	Exemplo
idWorkflow	Int	11	Chave Primária	Não pode ser nulo; Auto incremento	1
idTipoWorkflow	Int	11	Chave Estrangeira	Chave estrangeira da tabela “Tipo <i>workflow</i> ”	1
existeExame	Varchar	45	-	-	“Sim”
exameSuficiente	Varchar	45	-	-	“Não”
submeterExame	Varchar	45	-	-	“Sim”
aceitarExame	Varchar	45	-	-	“Não”
justificarRecusa	Varchar	255	-	-	“Utente não possui capacidade física para realizar o exame”
numUtente	Varchar	25	-	Número de utente do Doente a que corresponde o <i>workflow</i>	“269002355”
caseNumber	Varchar	45	-	Número do Caso executado no <i>workflow</i> .	“171”

Tipo Exame

Contém os tipos de exames que podem ser criados na BD.

Tabela 11 – Entidade Tipo Exame

Atributo	Tipo de Dados	Limite	Característica	Observações	Exemplo
idTipoExame	Int	11	Chave Primária	Não pode ser nulo; Auto incremento	1
designacao	Varchar	45	-	Não pode ser nulo	“Eletrocardiograma”

Utente

Contém a informação relativa aos Utentes do CHP.

Tabela 12 - Entidade Utente

Atributo	Tipo de Dados	Limite	Característica	Observações	Exemplo
idUtente	Int	11	Chave Primária	Não pode ser nulo; Auto incremento	1
nome	Varchar	255		Nome do utente; Não pode ser nulo	“Tiago Ribeiro”
numUtente	Varchar	25	-	Não pode ser nulo	“269002341”
idade	Int	11	-	-	24
sexo	Varchar	45	-	-	“Masculino”
peso	Varchar	45	-	-	“75”
alergias	Varchar	45	-	-	“Pólen”
pacemaker	TinyInt	1	-	Verifica se Utente possui <i>pacemaker</i> .	1

Exame

Permite consultar e inserir as informações relativas aos exames.

Tabela 13 - Entidade Exame

Atributo	Tipo de Dados	Limite	Característica	Observações	Exemplo
idExame	Int	11	Chave Primária	Não pode ser nulo; Auto incremento	1
idUtente	Int	11	Chave Estrangeira	Chave estrangeira da tabela "Utente"	1
idTipoExame	Int	11	Chave Estrangeira	Chave estrangeira da tabela "Tipo Exame"	1
frequencia	Int	11	-	-	1000
aquisicao	Int	11	-	-	11
frequenciaCardiograma	Int	11	-	-	70
duracaoP	Int	11	-	-	99
duracaoQRS	Int	11	-	-	82
duracaoT	Int	11	-	-	287
intervaloPR	Int	11	-	-	157
intervaloQT	Int	11	-	-	399
intervaloQTc	Int	11	-	-	431
eixoP	Varchar	45	-	-	"55.1"
eixoQRS	Varchar	45	-	-	"4.5"
eixoT	Varchar	45	-	-	"25.1"
rv5/sv1	Varchar	45	-	-	"1.31/0.76"
rv5+sv1	Varchar	45	-	-	"20.07"
dataPublicacao	DateTime	-	-	Não pode ser nulo; Data da publicação do exame	2018-11-15 17:00:00

Atributo	Tipo de Dados	Limite	Característica	Observações	Exemplo
comentarioMedico	LongText	4294967295	-	Contém o comentário médico relativo ao exame efetuado	“Estudo efetuado em modo espiral, com baixa dose de radiação...”
motivoExame	Varchar	255	-	Motivo para o pedido de exame; Não pode ser nulo	“Doente queixou-se de dor nas costas”

4.3.2 *ProcessMaker*

Nesta secção, o objetivo passa por apresentar alguns dos aspetos fundamentais da ferramenta BPM *ProcessMaker*, como as configurações que um utilizador com perfil administrador pode alterar na plataforma e as modificações realizadas no âmbito deste Protótipo. O segundo aspeto focado está relacionado com o *design* de processos, onde são apresentados alguns elementos da modelação de processos e os objetos que o *ProcessMaker* disponibiliza para criação. De notar que a instância de um Processo é denominada Caso ao longo do documento, dado ser essa a notação utilizada pelo *ProcessMaker*.

4.3.2.1 Configurações de administrador

Para o desenvolvimento do Protótipo é necessário proceder a algumas configurações no *ProcessMaker*, alterações essas que podem ser realizadas por um utilizador com o perfil administrador. Na Figura 10 é possível observar a *interface* para o utilizador com esses privilégios.

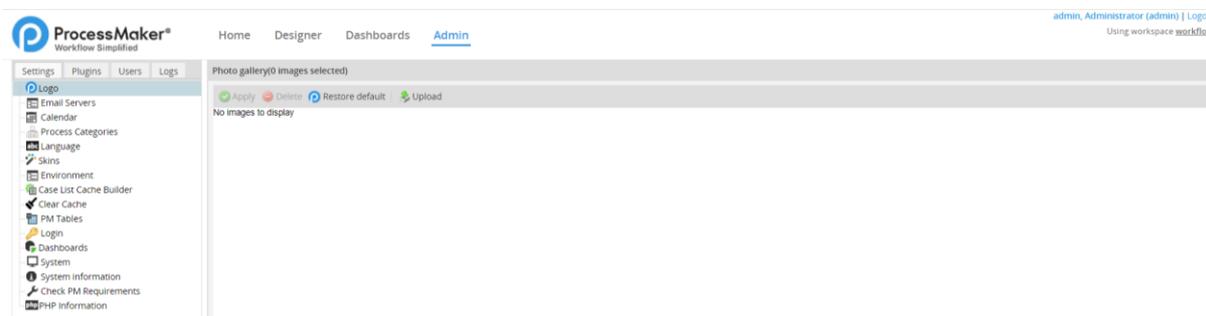


Figura 10 - Interface Administrador *ProcessMaker*

Como a imagem demonstra, um utilizador do tipo administrador pode modificar diversas definições da plataforma, como o logo que pretende utilizar, a forma como o calendário aparece para os

utilizadores, modificar a língua da ferramenta, entre outros. Este utilizador também pode gerir os *plugins* a utilizar no *ProcessMaker*, os utilizadores da plataforma e, por fim, aceder aos *logs* do sistema.

No contexto deste Protótipo, foi necessário configurar um servidor de Email, de forma a permitir o envio de mensagens eletrónicas. Esta configuração possibilita a utilização de alguns Eventos no *design* de processos que serão melhor explicados no ponto seguinte. Na Figura 11 consegue-se visualizar a configuração de Email utilizada neste Protótipo, de referir que os endereços eletrónicos estão riscados porque foram utilizados emails pessoais de forma a facilitar o teste da funcionalidade.

Email Engine	Server	Port	Sender Account	Sender Email	Sender Name	Use Secure Connection	Mail to	Default
PHPMAILER	smtp.gmail.com	587	[REDACTED]	[REDACTED]	Tiago Ribeiro	tls	[REDACTED]	✓

Figura 11 - Configuração do Servidor de Email

Outra configuração efetuada na plataforma está relacionada com os utilizadores que podem aceder ao *ProcessMaker*. De forma a contemplar as necessidades dos *workflows*, é essencial criar os utilizadores necessários para permitir que estes possam ser atribuídos a tarefas. Na criação destes utilizadores, é obrigatório introduzir as credenciais que estes devem utilizar para aceder à plataforma. Na Figura 12 é possível observar os utilizadores criados no *ProcessMaker* no âmbito deste Protótipo.

User Name	Full Name	Status	Role	Last Login	# Cases	Due Date
admin	admin, Administrator (admin)	Active	System Administrator	2018-07-30 22:00:47	0	2020-01-01 00:00:00
Tecnico	Matias, Luis (Tecnico)	Active	Operator		2	2019-08-24 00:00:00
Patient	Patient, Patient (Patient)	Active	Operator	2018-07-20 23:24:50	14	2019-06-03 00:00:00
HP	Ribeiro, Tiago (HP)	Active	Operator	2018-07-16 23:10:54	19	2019-06-03 00:00:00

Figura 12 - Utilizadores da Plataforma

Neste caso, a lista de utilizadores é composta pelo Administrador, que é criado automaticamente aquando da instalação da plataforma, um Técnico, um Profissional de Saúde e um Paciente. Esta lista pode ser alterada de acordo com as necessidades dos *workflows* que vão sendo criados. Referir ainda a possibilidade de criação de grupos, que permitem a associação de vários utilizadores e a alteração do estado dos utilizadores entre ativo, inativo e de férias. É importante mencionar que serão criados os utilizadores necessários para suprir as necessidades dos *workflows* criados no contexto deste Protótipo, logo não está contemplado o conjunto total de utilizadores que seriam necessários no contexto de todos os *workflows* clínicos no Centro Hospitalar do Porto. Na imagem também é possível visualizar a associação de um perfil a um utilizador. Basicamente, o *ProcessMaker* permite a criação destes perfis com o objetivo de os associar a utilizadores e definir as permissões que estes possuem na plataforma. No contexto deste Protótipo, os utilizadores foram criados com o perfil designado Operador, que permite,

entre outras definições, a criação de Casos e o *Login* no *ProcessMaker*. Estas permissões foram suficientes para o desenvolvimento do Protótipo e não houve necessidade de alterações.

4.3.2.2 *Design* dos Processos

Depois de uma explicação das configurações necessárias realizadas no *ProcessMaker*, inicia-se o esclarecimento de alguns aspetos importantes na construção dos Processos, de forma a facilitar o entendimento da plataforma.

O *ProcessMaker* utiliza a notação *Business Process Model and Notation 2.0* na modelação, de forma a que todos os *Stakeholders* do negócio possam assimilar a informação contida nos Processos mais facilmente. Nos *Stakeholders* incluem-se os analistas que criam e aperfeiçoam os processos, os técnicos responsáveis por os implementar e os gestores que devem monitorizar e administrar esses mesmos processos. Dito isto, o foco neste ponto do documento encontra-se nas particularidades do *ProcessMaker* na aplicação da notação BPMN 2.0 e como os elementos da notação podem ser configurados na plataforma, de maneira a facilitar a criação e leitura dos Processos. Na Tabela 14 encontra-se a lista de elementos que constituem a notação BPMN 2.0 presentes no *ProcessMaker*.

Tabela 14 - Elementos BPMN 2.0 do *ProcessMaker*

Elementos BPMN 2.0	
Eventos	<ul style="list-style-type: none"> • Evento inicial: <ul style="list-style-type: none"> ○ Vazio; ○ Receção de mensagem; ○ Temporizador; ○ Condicional; ○ Sinal. • Evento intermédio: <ul style="list-style-type: none"> ○ Mensagem de email; ○ Enviar mensagem; ○ Sinal. • Evento final: <ul style="list-style-type: none"> ○ Vazio; ○ Mensagem de email; ○ Mensagem; ○ Erro; ○ Sinal; ○ Concluir.
<i>Gateways</i>	<ul style="list-style-type: none"> • <i>Gateway</i> exclusivo; • <i>Gateway</i> inclusivo; • <i>Gateway</i> paralelo.
Tarefas	<ul style="list-style-type: none"> • Tarefa vazia;

Elementos BPMN 2.0	
	<ul style="list-style-type: none"> • Tarefa de envio; • Tarefa de receção; • Tarefa de utilizador; • Tarefa de serviço; • Tarefa de <i>script</i>; • Tarefa manual; • Tarefa regra de negócio; • Sub-Processo.
Base de dados	<ul style="list-style-type: none"> • <i>Data object</i>; • <i>Data store</i>.
Agrupamentos	<ul style="list-style-type: none"> • <i>Black Box Pool</i>; • <i>Pool</i>; • <i>Lane</i>; • Grupo.
Elementos de texto	<ul style="list-style-type: none"> • Anotação de texto; • Laço.

O *ProcessMaker* também permite a criação de objetos para cada processo. Na Tabela 15 é possível observar a diversidade de objetos que podem ser inseridos durante o *design* dos *workflows*.

Tabela 15 - Objetos de Processo (Adaptado de ProcessMaker (2018))

Objeto	Função
Variável	As variáveis são criadas para cada Processo e podem ser utilizadas por qualquer objeto presente nesta tabela. Dito isto, na maioria dos casos são utilizadas para popular os campos nos formulários e ao longo do Processo em <i>triggers</i> e <i>templates</i> de email.
Tipo de mensagem	Este objeto permite o envio de informação entre dois Processos. A informação pode ser enviada através de elementos Evento de envio e receção de mensagens.
Formulário dinâmico	Um formulário dinâmico funciona como uma tela onde vários controlos como botões, caixas de texto, caixas de seleção, etc, podem ser facilmente colocados para criar um formulário atrativo e legível.
<i>Input</i> de documento	Objeto que permite ao utilizador carregar um ou vários documentos e anexá-los a um Caso (instância de um Processo). Estes ficheiros podem ser documentos de texto, folhas de cálculo, imagens, ou qualquer outro tipo de documento.

Objeto	Função
<i>Output</i> de documento	Este objeto permite a criação de ficheiros durante a execução dos Casos. O <i>Output</i> de documentos pode ser particularmente útil na criação de cartas, faturas ou recibos. Os ficheiros podem ser gerados em <i>PDF</i> ou documentos <i>Word</i> .
<i>Trigger</i>	<i>Script</i> em <i>PHP</i> que pode ser inserida num Processo. Um <i>trigger</i> pode ser “disparado” antes ou depois de um Passo ser executado, antes de um utilizador ser atribuído para trabalhar na próxima tarefa do Processo e antes ou depois de um Caso ser encaminhado para a próxima tarefa.
Tabela de <i>report</i>	Estas tabelas permitem a exportação de dados da BD do <i>ProcessMaker</i> , de maneira a que o acesso à informação seja possível através de <i>queries SQL</i> ou aplicações externas. Este objeto também permite a partilha de informação com ferramentas de <i>reporting</i> externas.
Ligação a Base de dados	Permite que o <i>ProcessMaker</i> se ligue a Base de Dados externas, permitindo que uma organização possa integrar a ferramenta BPM com outras aplicações de negócios que utilizam um sistema de armazenamento de dados.
<i>Template</i>	Os <i>templates</i> são ficheiros <i>HTML</i> utilizados como <i>templates</i> de email para enviar notificações. Estes ficheiros são carregados num diretório para cada Processo e estão disponíveis para serem selecionados na criação de notificações.
Ficheiro público	Este objeto permite o armazenamento de ficheiros que podem ser acedidos por <i>triggers</i> ou <i>URL</i> .
Permissão do processo	Por norma, apenas utilizadores a que os Casos estão atribuídos podem trabalhar neles e obter acesso aos objetos, mas este objeto concede, a utilizadores ou grupos, acesso de leitura aos objetos no Caso. As permissões do Processo pretendem oferecer o controlo sobre como os utilizadores acedem aos Casos e que informação podem observar em Tarefas específicas.
<i>Case tracker</i>	Permite que um utilizador externo visualize os <i>logs</i> do <i>ProcessMaker</i> e analise a progressão de um dado Caso. Esse utilizador externo pode verificar em que Tarefa se situa o Caso, quais as Tarefas já completadas, o histórico do Caso e o histórico de mensagens.

Objeto	Função
Supervisor do processo	Utilizadores que possuem privilégios especiais para vigiar e analisar os Casos de um Processo em particular. Ao contrário do objeto Permissão de Processo, um Supervisor pode ter acesso de leitura e escrita aos Casos. Este tipo de utilizador pode, entre outras permissões, visualizar todos os Casos de um Processo, modificar os dados nos formulários dinâmicos depois de serem submetidos e apagar ou submeter novamente documentos.

No contexto deste Protótipo, foi essencial a criação das variáveis necessárias para a execução dos *workflows*, dos formulários que deverão ser submetidos pelos utilizadores para avançar com os Casos, da ligação à BD externa, para permitir a conexão entre o *ProcessMaker* e o sistema de armazenamento de simulação do sistema PCE e de *triggers* para realizar pedidos à BD externa. Apesar do destaque dado a estes objetos, que foram aplicados em praticamente todos os *workflows* criados, isto não implica que os restantes não tiveram uso. O que define a aplicação dos restantes objetos são as particularidades de cada *workflow* e os requisitos associados.

4.3.3 Aplicação *web* de gestão de *workflows* clínicos

O objetivo desta subsecção passa por apresentar as funcionalidades desenvolvidas na aplicação e como pretendem responder aos requisitos do Protótipo. De seguida, encontram-se um conjunto de diagramas de sequência que pretendem facilitar o entendimento de como a aplicação *web* construída comunica com a ferramenta BPM e as mensagens trocadas entre ambos. Posteriormente, apresenta-se as *interfaces* criadas para a aplicação *web* e, por fim, o conjunto de páginas PHP que contém a lógica da aplicação desenvolvida.

4.3.3.1 Funcionalidades

No desenvolvimento da aplicação de gestão de *workflows* clínicos foi tido em consideração que as funcionalidades desenvolvidas deveriam responder aos requisitos delineados para o Protótipo construído nesta dissertação. Dito isto, na Tabela 16 encontram-se as funcionalidades desenvolvidas e quais os requisitos que pretendem cumprir.

Tabela 16 – Funcionalidades da aplicação de gestão de workflows clínicos

Funcionalidade	Requisito
Estrutura para permitir comunicar com o <i>ProcessMaker</i> .	<ul style="list-style-type: none"> • A ferramenta escolhida deve permitir que os <i>workflows</i> possam ser executados em aplicações externas; • O Protótipo deve ser flexível, ou seja, refletir as mudanças nos <i>workflows</i> que vão sendo efetuadas na ferramenta BPM.
Acesso à aplicação através de um formulário de <i>Login</i> . A informação inserida é validada de acordo com os utilizadores presentes no <i>ProcessMaker</i> .	<ul style="list-style-type: none"> • O Protótipo deverá possuir um controlo de utilizadores no acesso à aplicação.
Permitir ao utilizador visualizar as seguintes listas de Casos: <ul style="list-style-type: none"> • Por responder; • Por iniciar; • Participadas pelo utilizador. 	<ul style="list-style-type: none"> • O Protótipo deve permitir que o utilizador visualize a informação sobre os <i>workflows</i> em que trabalhou.
Permitir criar novos Casos, nos <i>workflows</i> a que o utilizador esteja atribuído na primeira tarefa e responder aos Casos quando são encaminhados para uma tarefa a que o utilizador está atribuído.	<ul style="list-style-type: none"> • A ferramenta escolhida deve permitir que os <i>workflows</i> possam ser executados em aplicações externas.
Executar os <i>workflows</i> na aplicação.	<ul style="list-style-type: none"> • A ferramenta escolhida deve permitir que os <i>workflows</i> possam ser executados em aplicações externas.
Estrutura para permitir mostrar formulários externos à aplicação e <i>ProcessMaker</i> .	<ul style="list-style-type: none"> • O Protótipo deve permitir a integração de formulários externos.
Apenas envia para o <i>ProcessMaker</i> as variáveis necessárias para executar o <i>workflow</i> .	<ul style="list-style-type: none"> • O Protótipo deve ser flexível, ou seja, refletir as mudanças nos <i>workflows</i> que vão sendo efetuadas na ferramenta BPM.
Permite gerar e armazenar documentos com informação relativa aos <i>workflows</i> executados na plataforma.	<ul style="list-style-type: none"> • O Protótipo deve permitir que o utilizador visualize a informação sobre os <i>workflows</i> em que trabalhou.

Mais à frente nesta secção, será dada uma explicação mais detalhada sobre como as funcionalidades estão desenvolvidas e a lógica envolvida.

4.3.3.2 Diagramas de Sequência

De forma a facilitar a compreensão das funcionalidades desenvolvidas, foram criados Diagramas de Sequência que demonstram as trocas de mensagens que ocorrem entre os diferentes objetos na aplicação. O primeiro Diagrama demonstra como os utilizadores podem aceder à aplicação de gestão de *workflows* clínicos, o segundo apresenta as mensagens trocadas na criação de Casos e, por fim, o terceiro está relacionado com a execução dos *workflows* clínicos.

1. Aceder à plataforma

O primeiro Diagrama de Sequência, presente na Figura 13, apresenta as interações existentes no acesso à plataforma. Este processo começa quando um utilizador introduz os dados de autenticação na *Interface* de “*Login da plataforma*”, sendo que esses dados são enviados para o *ProcessMaker* para verificar a validade do utilizador. Se a informação não for validada pela ferramenta BPM, então é apresentada uma mensagem de “*Dados Inválidos*” ao utilizador, caso contrário, o *ProcessMaker* devolve um *token* que irá ser necessário para consumir a sua API. Esse *token* é armazenado na aplicação, assim como o *username* do utilizador. Se o utilizador que se autenticou na plataforma for do tipo “*Profissional de Saúde*”, então é realizado um pedido à API do *ProcessMaker* para receber a lista de Casos associados ao utilizador e reencaminhá-lo para a *interface* “*Lista de Casos*”, caso o utilizador seja do tipo “*Utente*”, então este seleciona o Caso que pretende iniciar e é reencaminhado para o *Formulário Inicial* do Caso que iniciou.

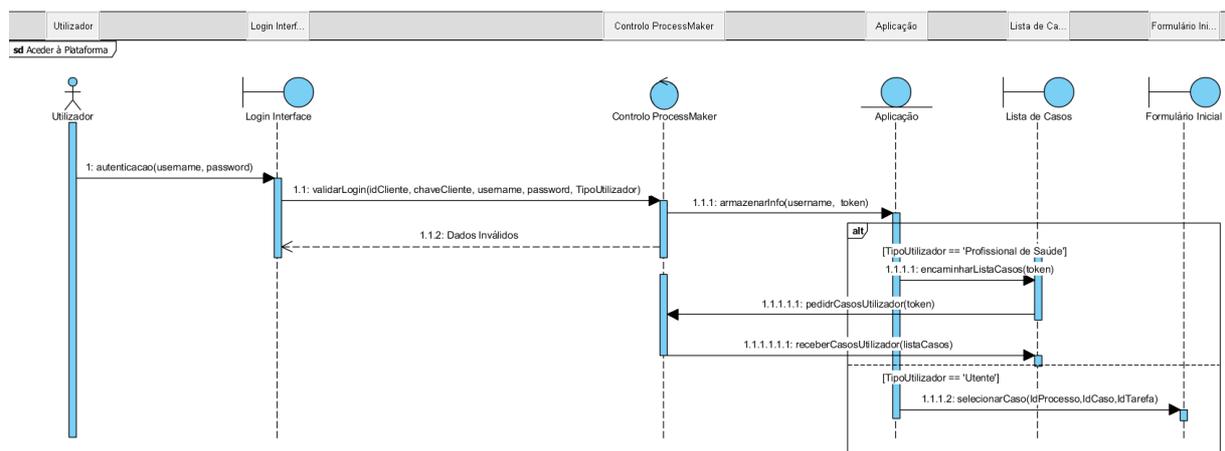


Figura 13 - Diagrama de Sequência: Aceder à Plataforma

2. Iniciar Caso

O Diagrama de Sequência “*Iniciar Caso*”, preocupa-se em demonstrar as comunicações entre os diferentes objetos quando um utilizador pretende criar um Caso. Esta sequência começa quando o

utilizador seleciona um Caso para ser iniciado, sendo que é enviado para o *ProcessMaker* o identificador do Processo e da Tarefa, variáveis necessárias para criar um Caso. Se o pedido for realizado com sucesso, a ferramenta BPM cria a instância do Processo e devolve o identificador do Caso criado, a localização do formulário associado à Tarefa e as variáveis que serão necessárias para o *ProcessMaker* continuar o *workflow*. A receção deste identificador e variáveis permite que a aplicação encaminhe o utilizador para o formulário inicial do Caso selecionado e, conseqüentemente, principiar a execução do *workflow* associado a esse Caso. O Diagrama aqui descrito pode ser visualizado na Figura 14.

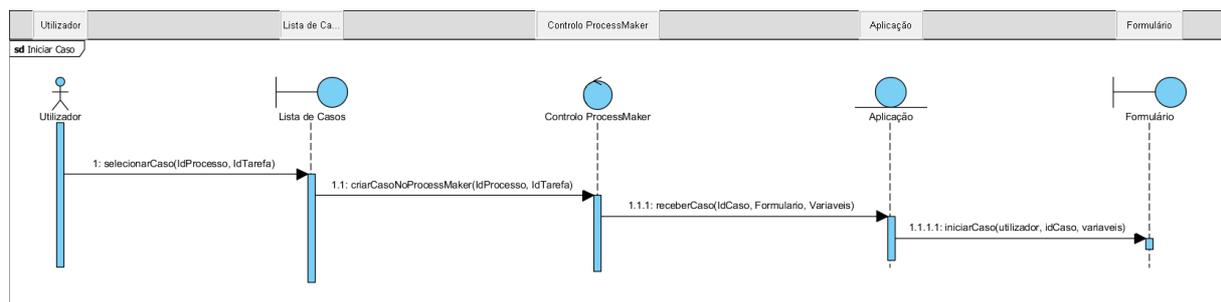


Figura 14 - Diagrama de Sequência: Iniciar Caso

3. Executar Workflow

De seguida, encontra-se o Diagrama de Sequência que representa a troca de mensagens que ocorrem na execução de *Workflows*. Um utilizador preenche o formulário e submete os dados inseridos, sendo que estes podem passar por um critério de validação caso existam regras associadas aos campos. A informação submetida para o *ProcessMaker* é constituída pelo identificador do utilizador que preencheu o formulário, o identificador do Caso que está a ser executado e as variáveis preenchidas no formulário que o *ProcessMaker* necessita para continuar a executar o fluxo. Após o processo de submissão de dados, o *ProcessMaker* envia para a aplicação a informação relativa à próxima Tarefa do *workflow*, de forma a perceber se o utilizador a que esta está atribuída é o mesmo que está autenticado na aplicação, caso essa condição seja verdadeira, a aplicação faz um pedido à ferramenta BPM para receber a informação do próximo formulário e encaminhar o utilizador para o mesmo, se a condição for falsa, o utilizador é encaminhado para a “Lista de Casos” e o *workflow* fica em pausa até o utilizador ao qual está atribuída a próxima Tarefa continuar o fluxo. Na Figura 15 é possível visualizar o Diagrama de Sequência que contém as interações acima descritas.

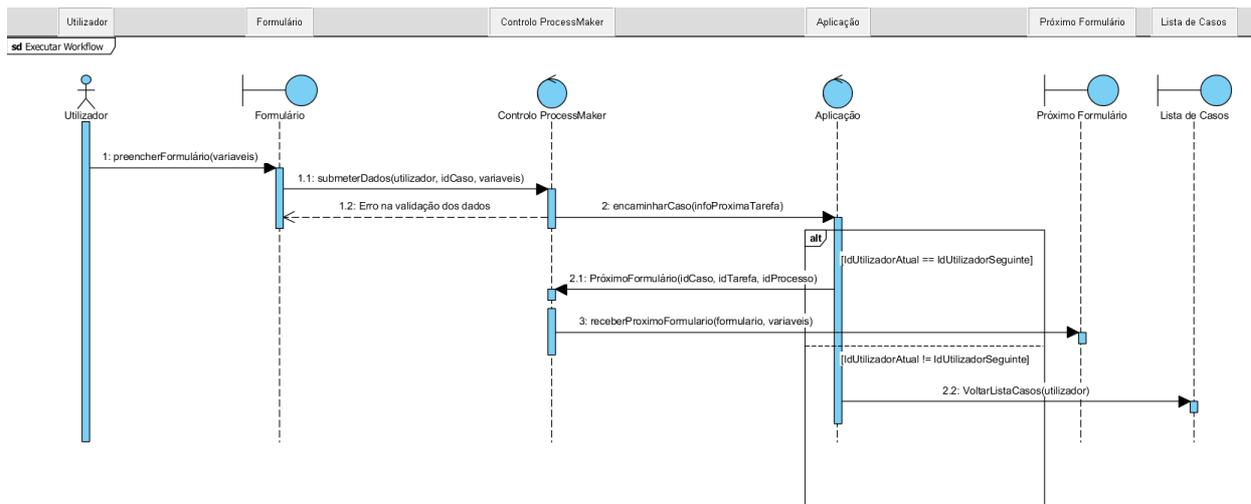


Figura 15 - Diagrama de Sequência: Executar Workflow

4.3.3.3 Interface

Neste ponto, o objetivo passa por mostrar o aspeto gráfico da aplicação de gestão de *workflows* clínicos. Primeiro são apresentadas as *interfaces* criadas para o acesso à plataforma por parte dos utilizadores, de seguida a *interface* “Lista de Casos” que contém, como o próprio nome indica a lista de casos que o utilizador pode aceder e visualizar depois de se autenticar na aplicação *web* e por fim um exemplo de um formulário recebido do *ProcessMaker* depois de criado um Caso.

Quando um utilizador entra na plataforma, pode escolher se quer utilizar a plataforma como Profissional de Saúde ou como Utente. Na Figura 16 é possível observar a *interface* que representa a interação.



Figura 16 - Interface “Tipo de Utilizador”

Caso o utilizador seleccione a opção “Entrar como Utente”, é-lhe imediatamente apresentado a lista de casos que pode iniciar. A aplicação foi construída desta maneira, depois de uma reunião com os responsáveis pelo desenvolvimento do sistema PCE, em que ficou decidido que não faria sentido criar um acesso no *ProcessMaker* para cada Utente, por isso foi atribuído um único acesso que todos os

Utentes partilham. De referir que caso existam *workflows* clínicos para os Utentes, estes serão identificados através do número de Utente que lhes será pedido no preenchimento dos formulários e verificado na Base de dados. Para proceder à iniciação do Caso selecionado pelo Utente, basta que o utilizador carregue no botão “Novo Caso”, como demonstrado na Figura 17.

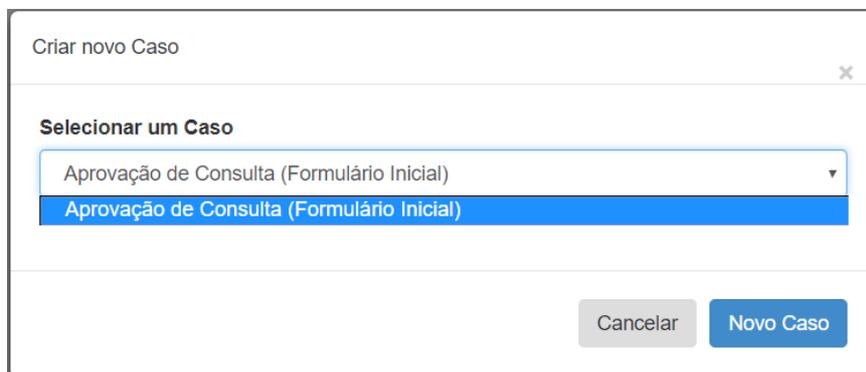


Figura 17 - Interface "Seleção de Caso" por parte do Utente

Caso o utilizador selecione a opção “Entrar como Profissional de Saúde”, é-lhe apresentado o formulário de autenticação da aplicação. Este formulário é necessário para os Profissionais de Saúde, ao contrário dos Utentes, porque cada um terá o seu acesso no *ProcessMaker*, tendo em conta que os *workflows* clínicos necessitam da envolvimento de diferentes especialistas, de acordo com as situações especificadas nesses mesmos *workflows*. Para aceder à aplicação, basta que o Profissional de Saúde preencha os seus dados de autenticação e pressione o botão “Login”. Na Figura 18 encontra-se o formulário referenciado neste parágrafo.

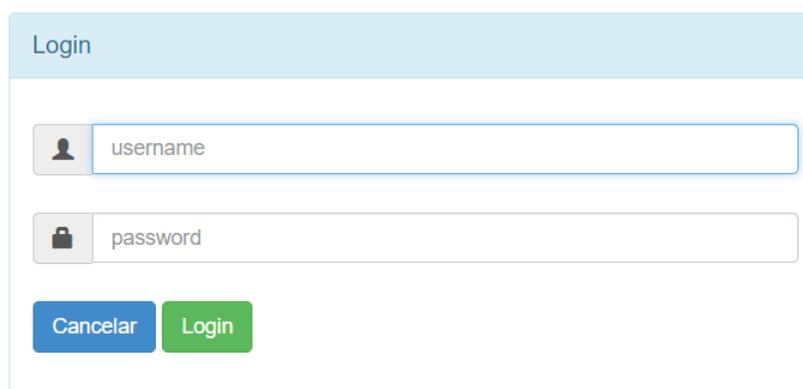


Figura 18 - Interface Login

Após a autenticação com sucesso do Profissional de Saúde, este é encaminhado para uma página nova que apresenta um conjunto de informações úteis para o utilizador. Entre essas informações encontra-se uma lista de Casos que o utilizador pode iniciar na aplicação, assim como acontece com o utilizador do tipo Utente, como já foi descrito acima. Mas as informações principais apresentadas e

únicas para os Profissionais de Saúde, são as tabelas disponibilizadas pelo *ProcessMaker* que permitem visualizar os Casos que estão a ser executados e o utilizador precisa de responder, os Casos que o utilizador criou, mas ainda não submeteu o formulário inicial e os Casos que o utilizador participou. Este conjunto de tabelas permite que o Profissional de Saúde possa gerir as instâncias dos Processos que necessitam do seu *input*, assim como guardar um histórico dos Casos em que participou. A *interface* retratada acima pode ser visualizada na Figura 19.

HP Log out

Selecione um Caso

Processo Exame TAC (Pedido d. ▾)

Novo Caso

Casos por responder (25)

#	Número do Caso	Processo	Tarefa	Enviado por	Estado	Data Inicio	Data de encaminhamento	
1	157	Processo Exame TAC	Recolher Variáveis	Ribeiro Tiago	TO_DO	2018-10-16 17:08:49	2018-10-16 17:08:59	Abrir
2	154	Processo Exame TAC	Recolher Variáveis	Ribeiro Tiago	TO_DO	2018-10-16 16:04:44	2018-10-16 16:04:53	Abrir
3	153	Processo Exame TAC	Recolher Variáveis	Ribeiro Tiago	TO_DO	2018-10-16 15:49:29	2018-10-16 15:49:39	Abrir
4	149	Processo Exame TAC	Verificar se exames são suficientes	Ribeiro Tiago	TO_DO	2018-10-16 02:46:37	2018-10-16 02:46:42	Abrir
5	148	Processo Exame TAC	Verificar se exames são suficientes	Ribeiro Tiago	TO_DO	2018-10-16 02:42:54	2018-10-16 02:43:00	Abrir
6	147	Processo Exame TAC	Verificar se exames são suficientes	Ribeiro Tiago	TO_DO	2018-10-16 02:40:31	2018-10-16 02:40:36	Abrir
7	146	Processo Exame TAC	Verificar se exames são suficientes	Ribeiro Tiago	TO_DO	2018-10-16 02:38:22	2018-10-16 02:38:26	Abrir
8	145	Processo Exame TAC	Verificar se exames são suficientes	Ribeiro Tiago	TO_DO	2018-10-16 02:34:54	2018-10-16 02:35:00	Abrir
9	144	Processo Exame TAC	Recolher Variáveis	Ribeiro Tiago	TO_DO	2018-10-16 02:33:08	2018-10-16 02:34:03	Abrir

Figura 19 - Interface Lista de Casos

Para finalizar, caso o utilizador decida criar um novo Caso ou abrir um que necessita de uma resposta sua, este é encaminhado para o formulário correspondente à Tarefa associada a esse Caso. Na Figura 20 pode-se observar um exemplo de um formulário.

Formulário Inicial

Preencha os seguintes dados

Nome	Idade
<input type="text"/>	<input type="text"/>
Email	Telemóvel
<input type="text"/>	<input type="text"/>
Pais	Tratamento
Portugal ▾	Dentário ▾

Próximo

Figura 20 - Interface Exemplo de formulário

4.3.3.4 Lógica da aplicação

A base lógica da aplicação ou *backend* está contruída num conjunto de páginas PHP, que recebem as decisões tomadas pelos utilizadores e estabelecem uma ligação com o *ProcessMaker* para comunicar as decisões tomadas pelos utilizadores e receber informação relativa ao estado das Tarefas e formulários.

Um aspeto importante de referir e que não é abordado diretamente nas descrições das páginas PHP está relacionado com a flexibilidade do Protótipo, ou seja, a forma como este deve refletir as mudanças efetuadas nos *workflows* desenvolvidos no *ProcessMaker*. Como a execução dos *workflows* no Protótipo é efetuada passo a passo, ou seja, quando uma tarefa termina, são realizados pedidos à API do *ProcessMaker* com o objetivo de encaminhar o Caso para a tarefa seguinte e recolher a informação relativa à mesma, significa que o Protótipo recebe sempre a versão atualizada do *workflow* presente na ferramenta BPM, logo permite refletir as mudanças efetuadas e executar o *workflow* de acordo com a modelação presente no *ProcessMaker*.

De seguida encontra-se a descrição das páginas desenvolvidas e a contribuição de cada uma para o funcionamento da aplicação.

1. *Login*

Esta página contém as funções responsáveis por controlar o acesso dos utilizadores à aplicação. Quando os utilizadores introduzem os dados de autenticação, estes são enviados para esta página de forma a serem validados. A página contém uma função chamada *pmRestLogin*, que pode ser visualizada [aqui](#), responsável por enviar os dados de autenticação para o *ProcessMaker* de forma a validar o utilizador. As variáveis “*username*” e “*password*” são inseridas pelo utilizador, enquanto que o “*client_id*” e “*client_secret*” estão definidos na página por se tratarem de variáveis fixas definidas pelo *ProcessMaker* e necessárias para utilizar a API da ferramenta. Caso o pedido de validação retorne o código HTTP 200, significa que existe um utilizador com as credenciais inseridas, então a ferramenta BPM envia um “*access_token*” que irá funcionar como um identificador do utilizador que está autenticado na aplicação sempre que for necessário fazer uma chamada à API. De referir que todas as chamadas à API, exceto esta inicial, necessitam desta variável, por isso o “*access_token*” é armazenado num *cookie* para utilização futura, assim como o “*username*”. Caso o código HTTP seja diferente de 200, o erro na validação é apresentado ao utilizador para este perceber o que deu errado no processo de autenticação.

Falta apenas referir uma interação neste processo. Como já indicado na apresentação das *interfaces*, a aplicação age de maneira diferente caso o utilizador seja um Profissional de Saúde ou Utente, isto deve-se a um controlo nesta página que reencaminha o utilizador para locais diferentes dependendo desse mesmo tipo. Os utilizadores do tipo Utente são reencaminhados para a lista de processos que podem iniciar, enquanto os Profissionais de Saúde são conduzidos para a sua Lista de Casos.

2. *CallProcessmakerEndpoints*

Esta página tem como principal função lidar com os pedidos à API do *ProcessMaker*. Para realizar esses pedidos, a página contém a função *pmRestRequest* que pode ser visualizada [aqui](#). Para executar esta função é necessário passar como parâmetros o método do pedido, o *endpoint* da API ao qual pretendemos realizar o pedido e, caso existam, o valor das variáveis que queremos enviar para a ferramenta BPM. Os métodos HTTP permitidos pela plataforma são o *Delete*, *Get*, *Post* e *Put*. Na Tabela 17 é possível o significado destes métodos e o que permitem realizar na API.

Tabela 17 - Métodos HTTP

Método	Significado
<i>Delete</i>	Permite eliminar um recurso.
<i>Get</i>	Permite receber um recurso.
<i>Post</i>	Permite inserir, atualizar e estender um recurso. alterar o estado de outros recursos e o envio de variáveis.
<i>Put</i>	Permite criar, atualizar, substituir um recurso e o envio de variáveis.

Caso o pedido retorne um código diferente de 200 ou 201, o erro é apresentado ao utilizador de maneira a oferecer alguma informação sobre o que aconteceu de errado na plataforma e para efeitos de *debug* no lado do desenvolvedor. Se o pedido for realizado com sucesso, é retornada a resposta da API do *ProcessMaker*. Referir ainda que a lista de *endpoints* é bastante alargada e permite utilizar grande parte das funcionalidades do *ProcessMaker* externamente.

Os *endpoints* podem ser do tipo Administração, *Design* e Caso. Os primeiros permitem alterar as definições e configurações da ferramenta e só podem ser aplicados por utilizadores com permissões de Administrador. Os *endpoints* de *Design* permitem modificar os *workflows* criados na plataforma, referir que só um utilizador com permissões pode consumir os *endpoints* deste tipo. Por fim, os *endpoints* mais utilizados são do tipo Caso, pois permitem manipular as variáveis de cada Caso, receber informação das tarefas que estão a ser executadas e encaminhar os Casos de acordo com o *workflow* que está a ser executado.

3. *ListCases*

Página responsável por apresentar a lista de Casos do Profissional de Saúde autenticado no sistema e permitir a continuação ou criação de Casos. Para a listagem dos Casos do utilizador foi criada uma função chamada *fillLists*, que pode ser visualizada [aqui](#). Esta função permite lidar com as diferentes tabelas disponibilizadas pelo *ProcessMaker* para gerir os Casos de cada Profissional de Saúde. A função

recebe dois parâmetros, a lista de Casos associados à tabela, sendo que as tabelas podem ser do tipo, “Por responder”, “Por iniciar” e “Participados pelo Profissional de Saúde”, e o título correspondente. A informação apresentada pelas tabelas é a seguinte:

- Número do Caso;
- Processo associado;
- Tarefa atual;
- Utilizador que encaminhou o Caso;
- Estado;
- Data de início do Caso (Apenas nas tabelas “Por responder” e “Por iniciar”);
- Data de encaminhamento do Caso (Apenas na tabela “Por responder”);
- Data de fim do Caso (Apenas na tabela “Participados pelo Profissional de Saúde”).

O estado dos Casos pode ser do tipo “Por responder”, caso o utilizador atual possua uma tarefa por concluir no Caso, “Por iniciar” se o Profissional de Saúde não submeteu o formulário inicial depois de criar o Caso, “em Pausa”, “Cancelado” e “Completado”. Referir ainda que as tabelas apenas são apresentadas se a lista de Casos para esse tipo for superior a zero e a tabela “Por responder” contém um botão que permite abrir o Caso na tarefa atual para continuar a execução do *workflow*.

Esta página também contém uma funcionalidade que permite aos Profissionais de Saúde iniciar Casos, sendo que o código que permite essa aplicabilidade pode ser visualizado nesta [ligação](#) e é executado quando o Profissional de Saúde seleciona o Caso que pretende criar. Este pedaço de código executa um pedido à API do *ProcessMaker*, utilizando o método “*Post*”, o *endpoint* relativo à criação de Casos e passando um conjunto de variáveis requeridas para essa mesma criação, sendo elas o identificador da Tarefa e o identificador do Processo. Depois de iniciada a instância do Processo, existe um encaminhamento para a página do *Dynaform*, junto com o envio das variáveis referidas na frase anterior e o identificador do Caso recebido aquando da criação do mesmo, de forma a iniciar o procedimento que irá permitir encaminhar o utilizador para o formulário inicial.

Referir ainda outra funcionalidade disponibilizada nesta página, que permite aos Profissionais de Saúde descarregarem a informação relativa aos Casos concluídos. O código que possibilita esta funcionalidade pode ser observado [aqui](#). Como se pode visualizar, a aplicação realiza um pedido à API do *ProcessMaker* para receber o documento gerado para o Caso em específico, e quando o obtém percorre um conjunto de funções PHP que permite armazenar o ficheiro na máquina local. Como referido, apenas quando os Casos estão concluídos é que esta opção está disponível para os Profissionais de Saúde, na Tabela “Casos participados pelo Profissional de Saúde”. Isto acontece porque os

documentos são gerados após a conclusão desses Casos, como será detalhado na descrição da página *RouteCase*.

4. *Dynaform*

A página *Dynaform* é responsável por reencaminhar os utilizadores para o formulário correspondente à Tarefa que está a ser executada no *workflow*. A parte do código que permite ao utilizador visualizar os formulários pode ser visualizado [aqui](#). A página *ListCases* reencaminha para a *Dynaform* os identificadores do Caso, da Tarefa e do Processo quando o utilizador inicia um Caso ou decide responder a um aberto, isto permite realizar uma primeira chamada à API para obter os “passos” da Tarefa. Os “passos” contêm a lista de *Triggers*, *Dynaforms*, *Input Documents* e *Output Documents* associados à Tarefa e como o objetivo passa por obter o formulário para o utilizador visualizar, então é lógico que este seja o primeiro passo. Depois desta primeira chamada, é obtido o identificador do formulário associado à Tarefa e procede-se à chamada seguinte para obter a informação do formulário. Para isso é necessário introduzir o identificador do formulário e do Processo. É importante notar que a API do *ProcessMaker* retorna os dados do formulário em formato *JSON*, por isso é necessário passar essa informação desse formato codificado para uma variável PHP.

Existiam dois requisitos pedidos pela equipa de desenvolvimento do sistema PCE que ainda havia a necessidade de serem tratados, a capacidade de encaminhar os utilizadores para formulários externos ao *ProcessMaker* e, apenas enviar para a ferramenta BPM as variáveis necessárias para executar o *workflow*. De forma a cumprir com esses requisitos, tornou-se impossível utilizar os formulários do *ProcessMaker* para apresentar visualmente ao utilizador. Dito isto, o caminho seguido passou por criar um formulário do *ProcessMaker* para cada Tarefa que necessita de um, sendo que esse formulário contém a informação relativa ao *link* do formulário externo e as variáveis que serão necessárias recolher para continuar a execução do *workflow*. Apesar de esta explicação poder parecer algo confusa, a imagem abaixo e respetiva explicação pode facilitar a compreensão desta funcionalidade.



Figura 21 - Exemplo Formulário ProcessMaker

Neste formulário do *ProcessMaker* existem duas variáveis escondidas, a que pode ser observada no lado esquerdo da Figura 21 contém o *link* para o formulário externo e será utilizada sempre que uma Tarefa necessitar de um formulário e no lado direito encontra-se uma variável necessária para continuar a execução do *workflow*, logo é necessário recolhê-la quando o formulário externo for submetido. De forma a que isso aconteça, o identificador, neste caso “*numUtente*”, deve estar presente como nome no *input* do formulário externo onde é inserido pelo utilizador, como por exemplo, “*<input name=“numUtente”/>*”.

Na Figura 22 é possível observar o encaminhamento do utilizador para o formulário, sem que este se aperceba do que está a acontecer. A variável “*\$linkToForm*” contém o *link* do formulário externo, as variáveis que contêm o utilizador autenticado na aplicação e o identificador do Processo são enviadas também para o formulário. Caso o formulário do *ProcessMaker* possua variáveis como o “*numUtente*”, necessárias para a execução do *workflow*, estas também são enviadas para o formulário externo.

```

<script type="text/javascript">
  function closethisasap() {
    document.forms["redirectpost"].submit();
  }
</script>
<body onload="closethisasap();">
<form method="post" name="redirectpost" action="<?=$linkToForm?>">
  <input type="hidden" id="logged-user" name="logged-user" value="<?=$variables['USER_LOGGED']?>" />
  <input type="hidden" name="app" value="<?=$_GET['app']?>" />
  <?php
  if(count($routeVariables) > 0){
  ?>
    <input type="hidden" name="variables" value="<?=$serializedArray?>" />
  <?php
  }
  ?>
</form>

```

Figura 22 - Reencaminhamento para formulário

5. RouteCase

Esta página é responsável por receber a informação dos formulários externos após a sua submissão e enviar esses dados para o *ProcessMaker* de forma a que se possa proceder ao encaminhamento dos Casos. O código desta página pode ser encontrado [aqui](#).

De forma a que o encaminhamento dos Casos seja possível, é necessário que nos formulários externos exista um pedido “*Post*” para a página *RouteCase*, em que seja incluída a informação relativa ao identificador do utilizador autenticado, o identificador do Caso que está a ser executado e as variáveis do formulário, abordadas no ponto anterior, caso existam. Ao ser identificado esse pedido e, caso o formulário envie alguma variável, é realizada uma solicitação à API do *ProcessMaker* para definir os valores das variáveis na ferramenta BPM através de um pedido “*Put*”. De seguida, são elaborados mais dois pedidos à API, o primeiro é novamente do tipo “*Put*” e irá efetuar o encaminhamento do Caso para

o passo seguinte no *workflow*. O segundo é um pedido “*Get*” para receber a informação relativa à Tarefa seguinte. Este último pedido é necessário, pois se o utilizador associado à próxima Tarefa for o mesmo que está autenticado no sistema, então este é encaminhado para o formulário seguinte, caso contrário, o utilizador autenticado já não é responsável por continuar o *workflow*, logo é encaminhado para a Lista de Casos.

Falta ainda destacar uma funcionalidade possibilitada pelo código presente nesta página e que permite gerar um documento com a informação do *workflow*. O código que permite esta funcionalidade pode ser observado [aqui](#). Quando o estado de um Caso passa para “*Completed*”, é escolhido o objeto do tipo *Output Document* que se pretende gerar e, após essa escolha, é realizado um pedido *Post* que permite criar o ficheiro, que fica associado ao Caso finalizado. Esta informação gerada pode ser descarregada pelos Profissionais de Saúde, como descrito na página *ListCases*.

CAPÍTULO 5. Implementação de um *workflow* clínico

Neste capítulo será apresentado o *workflow* clínico implementado no Protótipo. Esta implementação teve como objetivo testar se o Protótipo criado cumpre com os requisitos definidos, no contexto de um *workflow* real do Centro Hospitalar do Porto.

Nas duas primeiras secções deste capítulo, é realizada uma explicação detalhada da implementação do *workflow*, enquanto que na terceira é apresentada a avaliação de requisitos do Protótipo desenvolvido. A primeira secção possui a informação relativa à modelação do *workflow* na ferramenta BPM *ProcessMaker*. Nessa informação inclui-se a explicação lógica do *workflow*, a representação do mesmo utilizando a notação BPMN 2.0 e os objetos que foram necessários criar no *ProcessMaker*, de maneira a que o *workflow* fosse executado conforme a lógica definida. Na segunda secção, é possível observar a execução do *workflow* no Protótipo e a confirmação do cumprimento dos requisitos definidos na sua conceção. Na terceira secção pretende-se avaliar o cumprimento dos requisitos definidos na conceção do Protótipo e de que forma estes foram implementados.

Por fim, referir que o *workflow* representado neste capítulo foi disponibilizado e validado pelos elementos do CHP e orientadores desta dissertação.

5.1. Modelação do *workflow*

O objetivo desta secção passa por demonstrar o *workflow* de formulários clínico selecionado para testar a usabilidade e flexibilidade do Protótipo desenvolvido. Numa primeira fase, é possível observar a descrição do *workflow* escolhido, assim como a representação gráfica do mesmo utilizando a notação BPMN 2.0. De seguida, encontra-se o conjunto de objetos criados no *ProcessMaker* necessários para a correta execução do *workflow*.

De referir que no processo de seleção do *workflow* por parte dos elementos envolvidos, referidos no início deste capítulo, a principal preocupação foi garantir um certo nível de complexidade, que permitisse um seguimento de formulários dependente de decisões tomadas no preenchimento dos mesmos e a ligação à Base de dados de simulação do sistema PCE, de forma a demonstrar a conexão entre as diferentes plataformas.

5.1.1 Representação utilizando a notação BPMN 2.0

O *workflow* escolhido para testar o Protótipo passou pela representação do processo de pedido de um Exame de Tomografia Computorizada que ocorre no Centro Hospitalar do Porto. A representação gráfica deste *workflow*, utilizando a notação BPMN 2.0, pode ser visualizada na Figura 23.

O *workflow* é utilizado quando um Profissional de Saúde pretende iniciar o pedido de exame TAC para um Utente. Depois de introduzido o número de Utente, é necessário realizar uma verificação na Base de dados de forma a perceber se aquele Utente existe no sistema e, caso exista, possui exames armazenados. Caso o pedido à BD retorne informação, então o Profissional de Saúde pode visualizar os exames existentes para aquele Utente e decidir se são suficientes ou se é necessário proceder a análises adicionais. Na eventualidade do Profissional de Saúde considerar que os exames são suficientes, o *workflow* termina, caso contrário é necessário verificar um conjunto de dados do Utente, de maneira a perceber se está apto para a realização do exame TAC. Se o pedido à BD retornar que nenhum utilizador com aquele número de utente existe no sistema, então o Caso é encaminhado para a análise e recolha de variáveis, de forma a levantar os dados do Utente, perceber se está apto a realizar o exame e inseri-lo no sistema. No caso de o Utente existir, mas não possuir qualquer registo de exame na BD, então significa que será necessário proceder ao exame TAC, logo o *workflow* é encaminhado para a análise e recolha de variáveis, de maneira a que o Profissional de Saúde possa perceber se o Utente possui as condições necessárias para realizar o exame.

Nesta fase, o Profissional de Saúde decide se submete o pedido de exame. Se o pedido não for submetido, o *workflow* termina, caso contrário, o Caso é encaminhado para um Técnico que deverá proceder à realização do exame TAC. O Técnico pode recusar-se a fazer o exame, caso o faça, deve apresentar uma justificação que será enviada por email para o Profissional de Saúde que efetuou o pedido e o *workflow* termina. Na circunstância do Técnico decidir realizar o exame, então o Caso é novamente reencaminhado para o Profissional de Saúde que é responsável por publicar os resultados do TAC e, assim, finalizar o *workflow*.

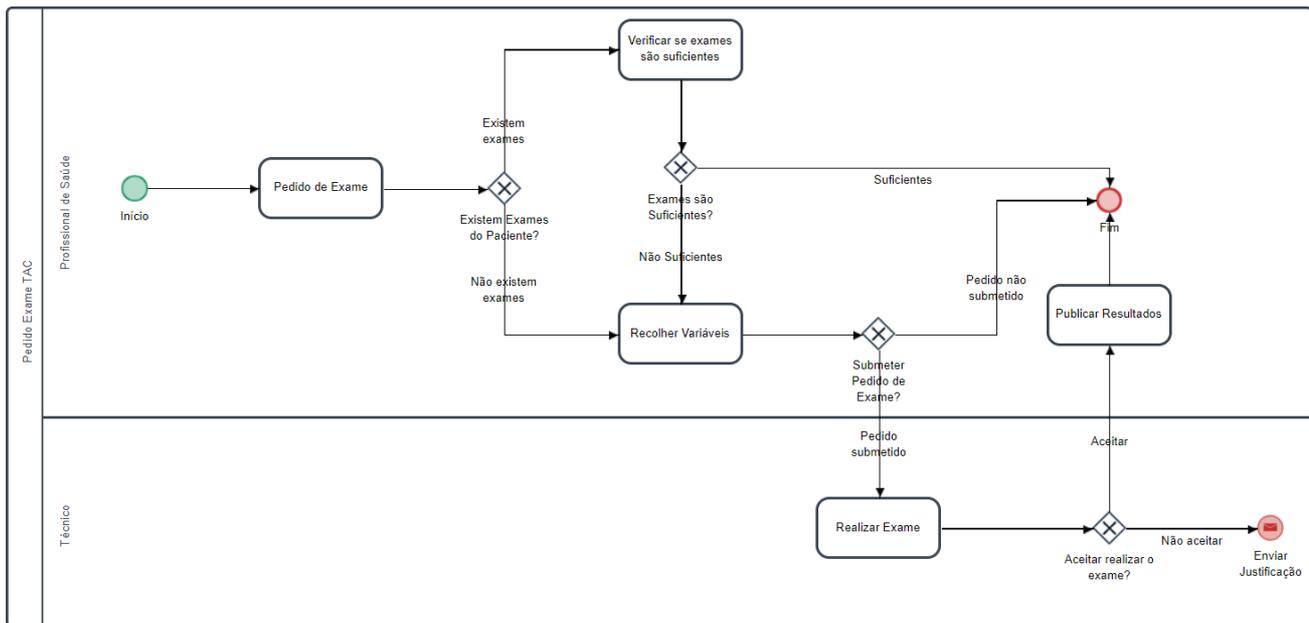


Figura 23 - Workflow "Pedido Exame TAC"

5.1.2 Objetos utilizados na modelação do *workflow*

De forma a que o *workflow* descrito pudesse ser executado corretamente, foi necessário criar um conjunto de objetos no *ProcessMaker*. Na Tabela 18 é possível observar as variáveis e *triggers* criados para cada Tarefa do *workflow*. De referir que para cada Tarefa foi criado um formulário com a mesma designação.

Tabela 18 – Variáveis e Triggers por Tarefa

Tarefa	Atribuição	Variável	Trigger
Pedido de exame	Profissional de Saúde	<ul style="list-style-type: none"> <i>numUtente</i> - armazena o número de Utente introduzido no formulário; <i>existeExame</i> – booleano que retorna verdadeiro se o Utente possuir exames no sistema ou falso caso não existam análises; <i>exames</i> – lista de exames que o utente possui registados no sistema; <i>utente</i> – informação relativa ao utente que está presente na BD de simulação do sistema PCE: <i>existeUtente</i> – booleano que verifica se o Utente, com o 	<u>Verificar utente e exames</u> - É executado antes do encaminhamento do caso para a próxima Tarefa. Permite verificar, através de chamadas à BD de simulação do sistema PCE, se o " <i>numUtente</i> " inserido existe no sistema e se possui exames guardados. As variáveis " <i>existeExame</i> " e " <i>existeUtente</i> " são preenchidas de acordo com o resultado obtido. A variável " <i>exames</i> " é preenchida com a lista de exames, caso existam, do Utente especificado. Caso exista um Utente com o número de utente especificado pelo utilizador, então a variável " <i>utente</i> " é preenchida

Tarefa	Atribuição	Variável	Trigger
		número de utente especificado, existe no sistema.	com a informação presente na BD relativa a esse mesmo Utente.
	Profissional de Saúde	<ul style="list-style-type: none"> • <i>caseNumber</i> – armazena o número do Caso; • <i>caseBeginDate</i> – guarda a informação relativa à data de início do Caso a ser executado; • <i>processName</i> – guarda o nome do Processo a ser executado 	Preencher variáveis para documento – Este <i>trigger</i> é executado após o encaminhamento da tarefa “Pedido de Exame”. O objetivo do <i>trigger</i> passa por armazenar informação que será utilizada quando o documento de informação do <i>workflow</i> for gerado.
Verificar se exames são suficientes	Profissional de Saúde	<ul style="list-style-type: none"> • <i>exameSuficiente</i> - guarda a decisão tomada pelo Profissional de Saúde em relação à suficiência dos exames que o Utente possui para caso que está a ser analisado. 	Guardar workflow – Este <i>trigger</i> é executado nas Tarefas em que as decisões tomadas podem levar à conclusão do <i>workflow</i> . Permite armazenar na BD de simulação do sistema PCE a informação relativa ao <i>workflow</i> executado.
Recolher variáveis	Profissional de Saúde	<ul style="list-style-type: none"> • <i>autorizarExame</i> – guarda a decisão do Profissional de Saúde, depois de analisar os dados do Utente, se autoriza e submete o pedido de exame. • <i>nomeHP</i> – permite armazenar a informação relativa ao nome do Profissional de Saúde que realizou o pedido de exame. • <i>emailHP</i> – permite guardar a informação do email do Profissional de Saúde que realizou o pedido de exame. 	Dados Profissional de Saúde – <i>Trigger</i> executado antes do encaminhamento para a tarefa seguinte. Permite guardar a informação das variáveis “ <i>nomeHP</i> ” e “ <i>emailHP</i> ”, através do uso de uma função interna do <i>ProcessMaker</i> . Esta informação é necessária para o envio de email caso o Técnico não aceite realizar o exame.
Realizar exame	Técnico	<ul style="list-style-type: none"> • <i>aceitarExame</i> – guarda a decisão do Técnico em relação à aceitação ou recusa em realizar o exame pedido pelo Profissional de Saúde. • <i>justificarRecusa</i> – caso o Técnico não aceite realizar o exame, deve justificar a recusa, de forma a que esta seja enviada por email para o Profissional de Saúde. • <i>nomeTecnico</i> – permite guardar a informação do nome do 	Dados Técnico – <i>Trigger</i> executado antes do encaminhamento para a próxima tarefa. Permite armazenar a informação do Técnico, através de uma função fornecida pelo <i>ProcessMaker</i> , para ser utilizada caso o Técnico recuse realizar o exame e precise enviar o email para o Profissional de Saúde. O elemento que permite enviar esse email caso o Técnico se recuse a realizar o exame pode ser visualizado aqui .

Tarefa	Atribuição	Variável	Trigger
		Técnico que tratou o pedido de exame.	
Publicar resultados	Profissional de Saúde	-	-

Na Figura 24 é possível observar o objeto criado para estabelecer a conexão com a BD de simulação do sistema PCE. Este objeto é necessário para os *triggers* “Verificar utentes e exames” e “Guardar *workflow*”.

Edit Database Connection ✕
 Engine:
 Encode:
 Server*:
 Database Name*:
 Username*:
 Password:
 Port*:
 Description:

Figura 24 - Objeto "Conexão à Base de Dados"

Também foi necessário criar um objeto do tipo *Output Document* para gerar o documento com a informação do *workflow*. Na Figura 25 é possível visualizar o conteúdo do objeto.

Edit Output Document ✕
 Workflow: @ #ProcessName
 Número do Caso: @ #caseNumber
 Data de Criação: @ #caseBeginDate
Número de Utente:
 @ #numUtente
Paciente Possui Exames?
 @ #existeExame
Exames Suficientes?
 @ #isExameSuficiente
Submeter pedido de exame?
 @ #autorizarExame
Exame realizado?
 @ #aceitarExame
Justificação para recusa:
 @ #justificarRecusa
 advanced.path: p Words:37

Figura 25 - Objecto Output Document "Informação do Caso"

5.2. Execução do workflow

Após a criação do *workflow* e todos os elementos envolventes no *ProcessMaker*, foi possível testar se o Protótipo é capaz de executar um *workflow* clínico real. Nesta secção é possível acompanhar a execução de um Caso do *workflow* “Exame TAC” na aplicação *web* de gestão de *workflows* clínicos criada com esse intuito. Notar que o Caso a ser demonstrado, contém um exemplo do *workflow* completo, ou seja, sem as decisões que terminam a sua execução a meio, de forma a não tornar a demonstração demasiado extensa e repetitiva. Também será possível visualizar uma variação do *workflow* real, de forma a comprovar a flexibilidade do Protótipo, ou seja, se este é capaz de refletir as modificações efetuadas nos *workflows* da ferramenta BPM.

5.2.1 Workflow real

Inicialmente, o Profissional de Saúde autentica-se na aplicação de gestão de *workflows* clínicos com as suas credenciais e acede à Lista de Casos que pode iniciar. Para iniciar um Caso do *workflow* descrito na secção anterior, o Profissional de Saúde apenas necessita de selecionar esse *workflow* na lista de opções, como se pode visualizar na Figura 26. Caso os leitores deste documento pretendam aprofundar e investigar os processos por trás da *interface*, é aconselhável visualizar a secção Lógica da aplicação, que detalha ao pormenor a lógica envolvida na aplicação.

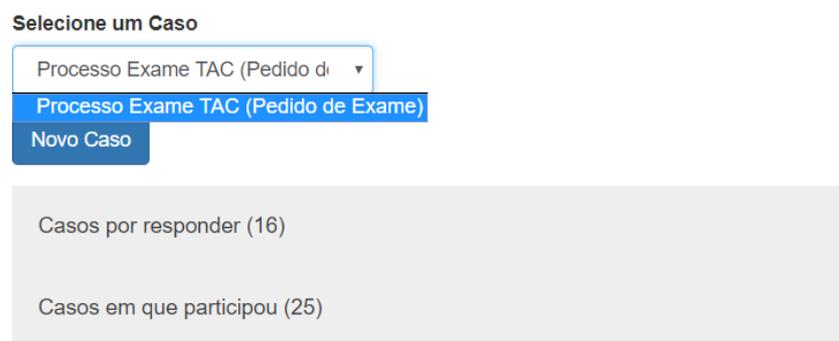


Figura 26 - Iniciar um Caso "Processo Exame TAC"

Após selecionar o botão “Novo Caso”, o utilizador é encaminhado para o formulário inicial do *workflow*, designado por “Pedido de Exame”. Este formulário pode ser visualizado na Figura 27.

Aqui, o Profissional de Saúde preenche o identificador do Utente. O identificador inserido é enviado para o *ProcessMaker*, após a submissão do formulário, e o *trigger*, que verifica se o Utente existe na BD de simulação do sistema PCE e se possui exames, é executado. Se o Utente existir e possuir exames na BD, então o Profissional de Saúde é encaminhado para o formulário “Verificar se exames são

suficientes”, caso contrário, passa à frente esse passo e é encaminhado diretamente para o formulário “Recolher Variáveis”. Como este exemplo de execução do *workflow* pretende demonstrar todos os passos e formulários, foi inserido um identificador de Utente que possui exames na BD de simulação do sistema PCE.

Pedido de Exame TAC



Figura 27 - Formulário "Pedido de Exame TAC"

Na Figura 28 é possível observar o formulário seguinte do *workflow* “Verificar se exames são suficientes”. Como o Utente referenciado no formulário anterior possui exames na BD, estes são apresentados para o Profissional de Saúde visualizar, neste caso em concreto, um TAC e um Eletrocardiograma. Depois de efetuar uma análise a esses exames, o Profissional de Saúde pode decidir se a informação que estes contêm é suficiente para tomar uma decisão em relação ao Utente e o problema que apresenta ou se os dados não são suficientes e é necessária informação adicional que pode ser obtida através de um exame TAC. Nas situações em que o Profissional de Saúde ache que a informação é suficiente, então, após a submissão do formulário, o Caso é encerrado. Se os dados forem considerados não suficientes, então o Profissional de Saúde é encaminhado para o formulário “Recolher Variáveis”.

Pedido de Exame TAC

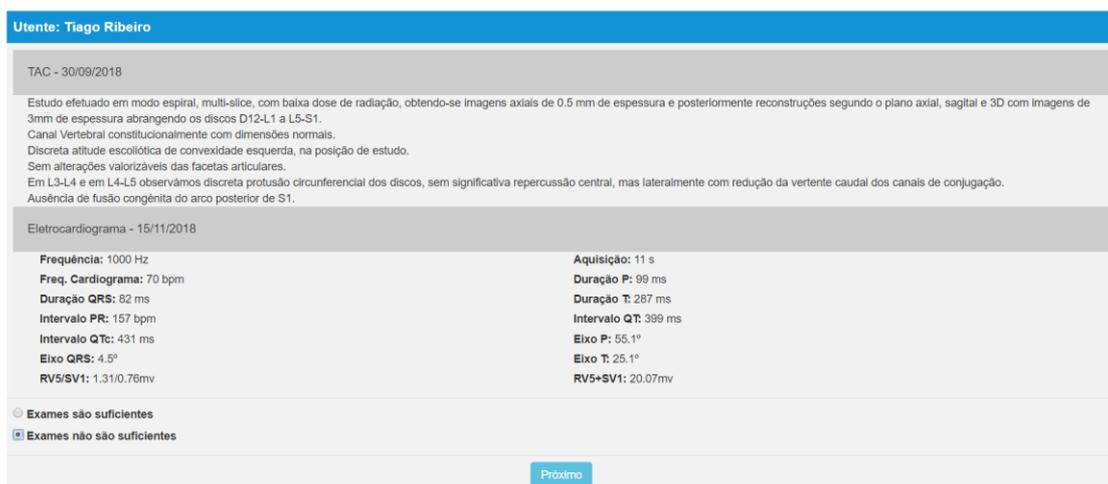


Figura 28 - Formulário "Verificar se exames são suficientes"

O Caso passa para o formulário “Recolher variáveis” que apresenta a *interface* visível na Figura 29, caso o Utente exista na BD de simulação do sistema PCE. Como é possível observar, os dados do Utente já estão inseridos, de forma a que o Profissional de Saúde os possa analisar, e decidir se o doente pode ser submetido ao exame TAC. Caso o faça, deve preencher o motivo para o pedido de exame, caso contrário deve justificar a razão para não submeter o pedido. Se o pedido de exame for submetido, o Profissional de Saúde recebe uma mensagem que o Caso foi reencaminhado para outro utilizador e a próxima Tarefa do *workflow* passa para o Técnico responsável por realizar o exame. Na eventualidade do pedido de exame não ser submetido, então o *workflow* termina a sua execução.

A *interface* nos casos em que o Utente não existe no sistema pode ser visualizada [aqui](#). Nestas situações, o Profissional de Saúde introduz as informações do Utente, mas necessita de decidir na mesma se o submete a exame ou não. Após a submissão do formulário, independentemente da decisão do Profissional de Saúde, o Utente é inserido na BD de simulação do sistema PCE.

Pedido de Exame TAC

Figura 29 - Formulário “Recolher variáveis”

De seguida, a continuação do *workflow* passa a estar dependente do Técnico responsável por realizar o exame TAC ao Utente. Este deve-se autenticar na plataforma com as suas credenciais, de forma a obter acesso à Lista de Casos que lhe estão atribuídos. Este processo pode ser observado na Figura 30.

Tecnico Log out

Casos por responder (4)

#	Número do Caso	Processo	Tarefa	Enviado por	Estado	Data Inicio	Data de encaminhamento	
1	171	Processo Exame TAC	Realizar Exame	Ribeiro Tiago	TO_DO	2018-10-17 01:25:15	2018-10-17 01:52:53	Abrir

Figura 30 - Técnico: Lista de Casos por responder

Após a seleção do botão “Abrir”, o Técnico é encaminhado para um formulário designado “Realizar Exame”, que pode ser observado na Figura 31. Neste formulário o utilizador preenche se

aceitou ou recusou realizar o exame. Se o Técnico se recusar a realizar o exame, então deve justificar e submeter a sua decisão, sendo que caso esta decisão seja tomada, o *workflow* termina e o Profissional de Saúde recebe um email com a justificação da recusa. O *template* de email pode ser observado [aqui](#). Caso o Técnico tome a decisão de aceitar realizar o exame, então o seu papel no *workflow* termina e o caso é encaminhado novamente para o Profissional de Saúde que fica responsável por publicar os resultados do exame. Neste exemplo, o Técnico aceitou realizar o exame e o caso foi encaminhado para o Profissional de Saúde.

Pedido de Exame TAC



Figura 31 - Formulário "Aceitar realizar exame"

Na última fase do *workflow*, o Profissional de Saúde autentica-se novamente na plataforma e, assim como o Técnico, seleciona o Caso em execução, na Lista de Casos por responder, de forma a continuar a execução do *workflow*. Após selecionar o botão "Abrir", é encaminhado para o formulário "Publicar Resultados" que pode ser observado na Figura 32. Este formulário, o último do *workflow*, permite que o Profissional de Saúde publique os resultados do exame TAC. Estes resultados são guardados na BD de simulação do sistema PCE, com o objetivo de armazenar a informação do exame relativa a aquele Utente em específico para que possa ser analisada e consultada no futuro. Após a submissão deste formulário, o utilizador recebe a mensagem que a sua parte na execução do *workflow* terminou e é encaminhado para a Lista de Casos.

Publicar Resultados

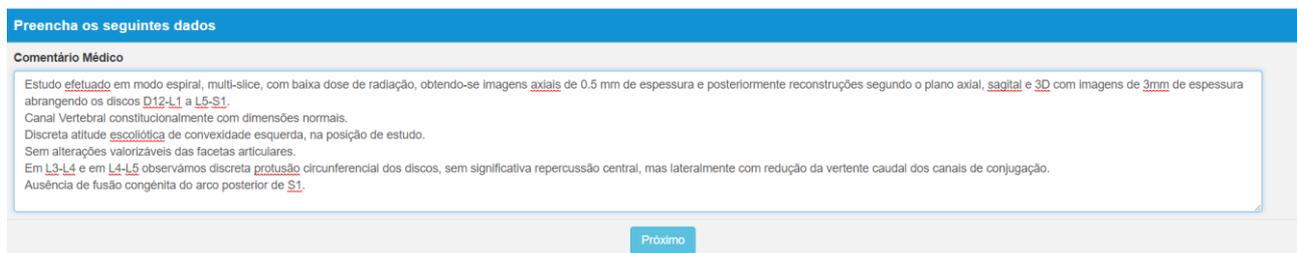


Figura 32 - Formulário "Publicar Resultados"

Na Figura 33, é apresentado o exame que foi criado na Base de dados de simulação do sistema PCE após o *workflow* seguido terminar. Também é possível visualizar [aqui](#), o formulário "Exames suficientes", caso um novo pedido de exame seja realizado para o mesmo Utente, onde já aparece o exame do *workflow* executado publicado.

	idExame	idUtente	idTipoExame	dataPublicacao	comentario	motivoExame
▶	18	1	1	2018-10-17 03:10:22	Estudo efetuado em modo espiral, multi-sli...	Doente queixou-se de dores nas costas.

Figura 33 - Exame criado na Base de dados

Após a conclusão do *workflow*, também foi gerado na BD um registo com as decisões tomadas ao longo da sua execução, como se pode observar na Figura 34. Relembrar que este registo foi criado através do *trigger* "Guardar *workflow*".

	idworkflow	idTipoWorkflow	existeExame	exameSuficiente	submeterExame	aceitarExame	justificarRecusa	numUtente	caseNumber
▶	4	1	Sim	Não	Sim	Sim		269005375	171

Figura 34 - Workflow criado na Base de dados

Por fim, referir que os Profissionais de Saúde também podem aceder a um documento que contém a informação das decisões tomadas ao longo do *workflow*. O exemplo do documento para o *workflow* executado nesta secção pode ser visualizado na Figura 35.

Workflow: Processo Exame TAC

Número do Caso: 171

Data de Criação: 2018-10-17 01:25:15

Número de Utente:
269005375

Paciente Possui Exames?
Sim

Exames Suficientes?
Não

Submeter pedido de exame?
Sim

Exame realizado?
Sim

Justificação para recusa:

Figura 35 - Documento gerado do workflow executado

5.2.2 Workflow modificado

A forma encontrada para testar a flexibilidade do Protótipo passou por modificar o *workflow* “Pedido Exame TAC” e executá-lo novamente para perceber se as alterações ficaram refletidas na aplicação *web*. Na Figura 36 é possível visualizar o *workflow* modificado, sendo que, apenas foi feita uma pequena alteração em relação ao original. Se o Técnico recusar realizar o exame ao Utente, então em vez de enviar um email ao Profissional de Saúde a explicar os motivos, o Caso é simplesmente reencaminhado para a tarefa “Recolher Variáveis”.

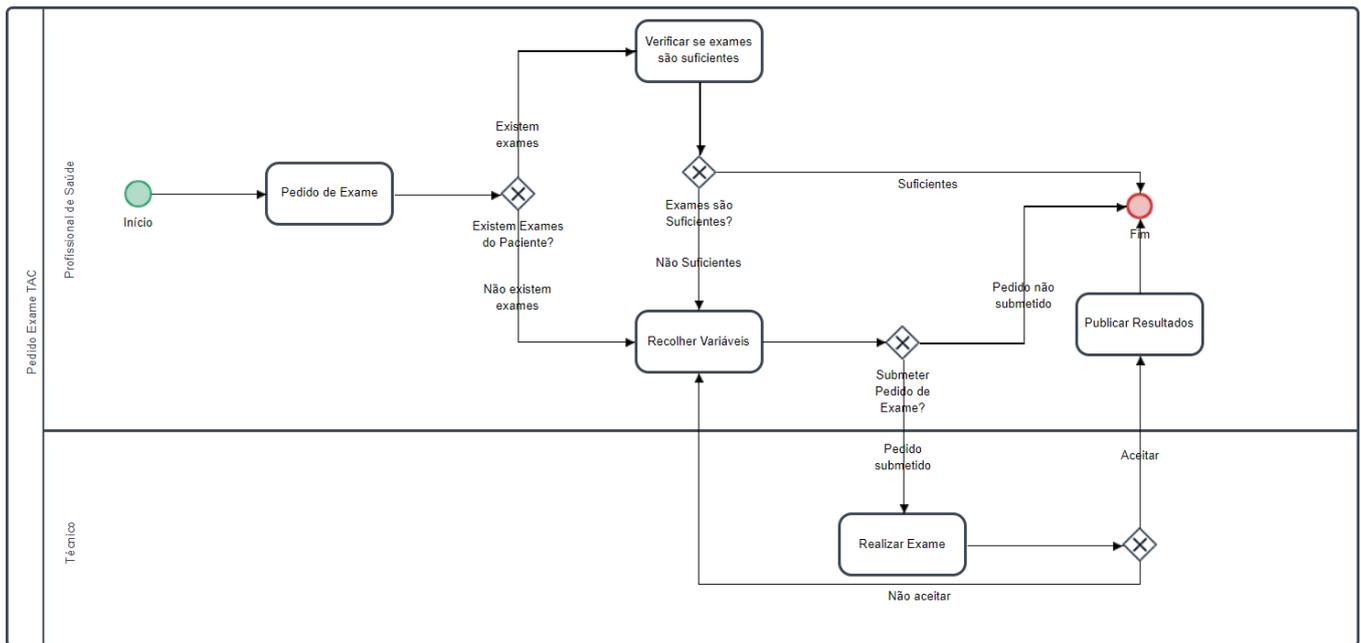


Figura 36 - Workflow "Pedido Exame TAC" modificado

Como o objetivo desta demonstração passou por provar que as mudanças nos *workflows* modelados no *ProcessMaker* são refletidos no Protótipo, apenas foram apresentados elementos que apresentem essas alterações, de forma a não repetir os formulários visíveis na demonstração anterior.

Na Figura 37 é possível visualizar que o Caso já se encontra na Tarefa “Realizar Exame”, ou seja, o Técnico é o responsável por continuar a sua execução. De referir que é importante estar atento ao número do Caso, neste caso 104, pois permite identificá-lo nas listas de Casos. Também é possível visualizar que foi o Profissional de Saúde que enviou o Caso para esta Tarefa, pois o Profissional de Saúde é representado pelo utilizador Tiago Ribeiro e Luís Matias representa o Técnico.

Casos por responder (1)						
#	Número do Caso	Processo	Tarefa	Enviado por	Estado	
1	104	Processo Exame TAC	Realizar Exame	Ribeiro Tiago	TO_DO	Abrir

Figura 37 - Técnico: Caso com workflow modificado

Na Figura 38 encontra-se o formulário “Realizar Exame”. Este formulário é preenchido pelo Técnico, que ao selecionar a opção “Não Aceitar”, recusa-se a realizar o exame, apresenta a justificação para tal decisão e submete o formulário.

Figura 38 - Formulário "Realizar Exame" workflow modificado

Como é possível observar na Figura 39, o Caso número 104 foi enviado pelo Técnico Luís Matias e encontra-se na Tarefa “Recolher Variáveis”, o que comprova que as alterações efetuadas no *workflow* “Pedido Exame TAC” no *ProcessMaker* se encontram refletidas no Protótipo, garantindo que este cumpre o requisito de flexibilidade.

Casos por responder (18)						
#	Número do Caso	Processo	Tarefa	Enviado por	Estado	
1	104	Processo Exame TAC	Recolher Variáveis	Matias Luis	TO_DO	Abriu

Figura 39 - Profissional de Saúde: Caso com workflow modificado

5.3. Avaliação dos requisitos do Protótipo

O Protótipo desenvolvido nesta dissertação necessitou de ser avaliado de forma a perceber se os requisitos definidos em conjunto com os elementos do Centro Hospitalar do Porto foram cumpridos. Na Tabela 19 é possível observar essa avaliação de requisitos.

Tabela 19 - Avaliação dos requisitos do Protótipo

Requisito	Resolução
Utilização de uma ferramenta, que permita a configuração e gestão de <i>workflows</i> clínicos	Este requisito foi cumprido através da utilização de uma ferramenta de <i>Business Process Management</i> que permite a gestão e configuração de <i>workflows</i> . No contexto desta dissertação, a ferramenta BPM permite configurar e gerir os <i>workflows</i> clínicos, mais especificamente, os formulários baseados no módulo <i>OpenEHR</i> .
A ferramenta escolhida deve ser gratuita	O <i>ProcessMaker</i> foi a ferramenta BPM selecionada para o Protótipo desenvolvido. A grande maioria das ferramentas BPM implicam custos para o utilizador, sendo que o <i>ProcessMaker</i> acabou por ser uma das poucas ferramentas encontradas que possuíam versões gratuitas. Nesse conjunto de ferramentas, o <i>ProcessMaker</i> reuniu o conjunto de

Requisito	Resolução
	<p>características mais apelativas para o Protótipo em questão. As explicações mais concretas para esta escolha podem ser observadas aqui.</p>
<p>A ferramenta deve permitir que os <i>workflows</i> possam ser executados em aplicações externas, de forma a assegurar a interoperabilidade do Protótipo</p>	<p>A importância da interoperabilidade do Protótipo deve-se à necessidade de conectar a aplicação que executa os <i>workflows</i> clínicos configurados com a ferramenta BPM, que contém essas configurações. O facto do <i>ProcessMaker</i> disponibilizar uma API, permitiu que a aplicação pudesse consumir essa API através de pedidos REST e recolher a informação relativa à configuração dos <i>workflows</i>. Essa comunicação, através da API, permitiu assegurar a interoperabilidade do Protótipo. Informações mais detalhadas sobre o consumo da API do <i>ProcessMaker</i> por parte da aplicação de execução dos <i>workflows</i> podem ser encontradas na subsecção <u>Lógica da aplicação</u>.</p>
<p>A ferramenta deve permitir a integração de base de dados externas</p>	<p>Na execução dos <i>workflows</i>, por vezes, é necessário que <i>ProcessMaker</i> consulte informação na base de dados do sistema Processo Clínico Eletrónico, como no caso do <i>workflow</i> clínico “Pedido Exame TAC”, em que foi necessário consultar se o Utente possuía exames. O <i>ProcessMaker</i> disponibiliza um objeto que permite estabelecer ligações com base de dados externas, sendo que esse objeto pode ser utilizado em qualquer <i>workflow</i>, permitindo assim, no contexto deste Protótipo, que o <i>ProcessMaker</i> conseguisse comunicar com a BD de simulação do sistema PCE. A lista de objetos disponibilizados podem ser consultados mais em pormenor na Tabela 15, enquanto que o objeto de ligação à BD criado no âmbito deste Protótipo pode ser visualizado na Figura 24.</p>
<p>O Protótipo deve permitir a integração de formulários externos, baseados no módulo <i>OpenEHR</i></p>	<p>Como os formulários utilizados no CHP são baseados nos arquétipos do módulo <i>OpenEHR</i> e disponibilizados na WEB, foi necessário criar uma estratégia que permitisse referenciar a hiperligação desses formulários no <i>ProcessMaker</i> de forma a que a aplicação pudesse proceder aos respetivos reencaminhamentos durante a execução dos <i>workflows</i>. Como a lógica por detrás deste requisito é relativamente complexa, sugere-se a leitura da explicação da página PHP <u><i>Dynaform</i></u>, para o</p>

Requisito	Resolução
	entendimento de como o Protótipo permite a integração de formulários externos.
O Protótipo deve ser flexível, ou seja, refletir as mudanças nos <i>workflows</i> que vão sendo efetuadas na ferramenta selecionada	A importância da flexibilidade do Protótipo deve-se às constantes mudanças dos <i>workflows</i> nos ambientes clínicos, surgindo a necessidade de facilitar essas modificações e que estas estejam refletidas na execução dos <i>workflows</i> . A primeira parte é assegurada pela ferramenta BPM que permite alterar facilmente os <i>workflows</i> , desde que exista um conhecimento mínimo da notação BPMN 2.0. A explicação da flexibilidade do Protótipo está diretamente relacionada com a interoperabilidade, pois como a aplicação comunica com o <i>ProcessMaker</i> através da API deste, sempre que uma Tarefa do <i>workflow</i> é encaminhada, a aplicação trata de recolher a informação da Tarefa seguinte e encaminhar o utilizador de acordo com os dados obtidos. Isto permite que a aplicação execute os <i>workflows</i> sempre de acordo com o que está presente no <i>ProcessMaker</i> , como se pode observar na secção anterior <u>Execução do workflow</u> , que, como o nome indica, apresenta a representação gráfica do <i>workflow</i> “Pedido Exame TAC” e uma versão alterada desse mesmo fluxo e é possível observar que o Protótipo reflete as alterações efetuadas na ferramenta BPM.
O Protótipo deve assegurar o controlo dos acessos à aplicação	O <i>ProcessMaker</i> possui uma funcionalidade de gestão de utilizadores que podem aceder à plataforma. De forma a que os diferentes tipos de utilizadores possam aceder à sua lista de Casos, foi necessário criar uma funcionalidade de autenticação na aplicação que, depois dos utilizadores inserirem os dados, pudesse comunicar com o <i>ProcessMaker</i> de forma a perceber se o utilizador existe na ferramenta BPM. Caso a resposta seja positiva, o utilizador tem acesso aos seus Casos, caso contrário, é apresentada uma mensagem de erro, de acordo com os dados fornecidos pelo utilizador. Uma explicação mais detalhada de como este requisito foi cumprido pode ser observada na explicação da página PHP <u>Login</u> .
O Protótipo deve possibilitar que os Profissionais de Saúde possam	O Profissional de Saúde, quando entra plataforma com as suas credenciais, pode visualizar uma Tabela, denominada “Casos participados pelo Profissional de Saúde”, que contém os <i>workflows</i> em

Requisito	Resolução
visualizar as informações sobre os <i>workflows</i> em que trabalharam	que o utilizador, em algum momento, participou numa Tarefa. Para além deste registo do histórico dos <i>workflows</i> , o Profissional de Saúde também tem a possibilidade de descarregar a informação detalhada relativa aos Casos finalizados. O documento contém as decisões tomadas ao longo do <i>workflow</i> , permitindo assim manter um registo documental que podem ser usados no futuro caso haja necessidade. A explicação de como o documento é gerado pode ser visualizada na descrição da página <u><i>RouteCase</i></u> e a informação de como o documento é descarregado pelos utilizadores pode ser observada na página <u><i>ListCases</i></u> .

Após a análise da avaliação realizada aos requisitos do Protótipo e a execução do *workflow* clínico real, é possível concluir que os requisitos definidos na conceção do Protótipo foram cumpridos. Apesar disto, será necessário implementar mais um conjunto alargado de testes, que incluem mais *workflows* clínicos reais, de forma a perceber se a plataforma possui *bugs* ou erros que não foram perceptíveis durante a implementação do *workflow* clínico “Pedido de exame TAC”.

CAPÍTULO 6. Conclusões

Neste capítulo é realizado um conjunto de apreciações sobre o trabalho desenvolvido ao longo desta dissertação. O capítulo contém uma análise aos objetivos definidos inicialmente na investigação, a avaliação se o conjunto de resultados esperados foram atingidos e alguns comentários em relação ao trabalho futuro do tema em análise nesta dissertação.

6.1. Análise dos Objetivos

No Capítulo 1 deste documento, foi referido que o principal objetivo desta dissertação passava por investigar se o modelo *OpenEHR* poderia ser aplicado na configuração e gestão de *workflows* clínicos, sendo que o foco passaria pelos formulários clínicos baseados em arquétipos *OpenEHR*. De forma a dar resposta a esse objetivo, numa fase inicial, foi necessário realizar um enquadramento conceptual do estado atual dos *workflows* clínicos, do modelo *OpenEHR* e como os conceitos se relacionam e compatibilizam para permitir a melhoria da atuação clínica. Com essa finalidade em mente, foi realizado um estudo dos trabalhos realizados sobre o tema em questão, de maneira a facilitar a aprendizagem sobre os principais paradigmas e termos associados ao tema, assim como trabalhos experimentais já elaborados na área com características semelhantes ao Protótipo obtido. Estas análises possibilitaram a realização de um Protótipo como pretendido na fase inicial da dissertação.

A solução desenvolvida permite a configuração dos *workflows* clínicos através de uma ferramenta BPM e a sua execução através de uma aplicação criada com esse intuito. Para isso foi necessário assegurar a interoperabilidade entre a aplicação e a ferramenta BPM *ProcessMaker*, de forma a possibilitar a comunicação entre os dois sistemas, a flexibilidade para permitir que as configurações efetuadas no *ProcessMaker* fossem refletidas durante a execução do *workflow* na aplicação e garantir a possibilidade dos formulários clínicos, baseados em arquétipos do módulo *OpenEHR*, serem chamados externamente à aplicação e ferramenta BPM. Para além destes requisitos delineados para o Protótipo, a aplicação também permite que os Profissionais de Saúde possam aceder e guardar a informação relativa aos *workflows* que participaram. Tendo isto em conta, o Protótipo desenvolvido permite resolver um dos problemas encontrados no Centro Hospitalar do Porto, ao garantir a integração dos *workflows* de formulários clínicos no sistema Processo Clínico Eletrónico, tornando mais simples o processo de modificação dos processos existentes no hospital. Como o Protótipo permite a configuração e gestão dos *workflows* clínicos, tendo como foco os formulários baseados arquétipos do modelo *OpenEHR*, então

este trabalho permitiu concluir que, de facto, é possível a aplicação desse modelo na configuração e gestão de *workflows* clínicos. De referir que o Protótipo desenvolvido foi sendo validado com um elemento da equipa de desenvolvimento do sistema PCE no CHP, garantindo que a solução estaria de acordo com o pretendido, de forma a suprimir as necessidades no centro hospitalar.

6.2. Resultados atingidos

O trabalho realizado durante esta dissertação englobou a conceção, desenvolvimento e validação de um artefacto que permite a configuração e gestão de *workflows* clínicos. O foco do artefacto esteve nos formulários baseados em arquétipos do modelo *OpenEHR*. O artefacto desenvolvido permite complementar o sistema PCE desenvolvido no CHP, englobando os *workflows* clínicos que se encontram em falta nesse sistema. Outro problema que existe atualmente no sistema PCE é a dificuldade em modificar os *workflows* clínicos. Com este artefacto, basta um conhecimento mínimo da notação BPMN 2.0 e da ferramenta BPM *ProcessMaker* para proceder às modificações necessários dos processos, oferecendo aos Profissionais de Saúde mais possibilidades e flexibilidade na execução das suas tarefas clínicas.

Apesar desta dissertação ter implicado o desenvolvimento de um Protótipo, também é importante realçar os contributos para a comunidade científica. Num mundo atual em que a globalização de sistemas é cada vez uma realidade maior, surge o modelo *OpenEHR* que tem como objetivo assegurar a interoperabilidade universal dos dados eletrónicos clínicos. Esta dissertação aborda a forma como o modelo *OpenEHR* pode ser aplicado nos *workflows* clínicos através do uso de uma ferramenta BPM, de maneira a melhorar os processos desempenhados diariamente pelos Profissionais de Saúde e, conseqüentemente, a atuação clínica. O artefacto desenvolvido foi a forma encontrada para dar resposta a este problema.

De realçar que foi publicado um artigo científico, sobre o tema abordado nesta dissertação, que visou abordar os contributos para a comunidade científica que a investigação efetuada produziu.

6.3. Trabalho futuro

Como se prevê que o artefacto desenvolvido seja implementado no CHP, através da sua integração no sistema PCE, de forma a ser utilizado pelos Profissionais de Saúde e facilitar as modificações dos *workflows* clínicos existentes no centro hospitalar, é necessário limar algumas arestas

antes dessa implementação. Como o Protótipo será utilizado pelos Profissionais de Saúde, então será necessário que os mesmos testem a aplicação obtida, de forma a perceber se responde na integra às necessidades em relação ao sistema atual ou se existe alguma questão que falta abordar. Como o meio de contacto foi sempre a equipa de desenvolvimento do sistema PCE do CHP, nunca houve um *feedback* por parte dos Profissionais de Saúde em relação ao Protótipo desenvolvido. Outra questão em relação ao Protótipo prende-se com a necessidade de uma bateria de testes mais alargada de forma a garantir que a aplicação executa todas as variações de *workflows* clínicos reais com sucesso.

Por fim, aconselha-se a continuação do estudo da aplicação do modelo *OpenEHR* nos *workflows* clínicos através do uso de ferramentas BPM, devido ao elevado número de benefícios que podem advir dessa aplicação, não só nos *workflows* de formulários baseados em arquétipos *OpenEHR*, como esta investigação se focou, mas também em outras áreas de atuação clínica.

Referências

- Aagesen, G., & Krogstie, J. (2015). Bpmn 2.0 for modeling business processes. In *Handbook on Business Process Management 1: Introduction, Methods, and Information Systems*. https://doi.org/10.1007/978-3-642-45100-3_10
- ArisPlatform. (2017). ARIS Express details. Retrieved from <http://www.ariscommunity.com/aris-express/details>
- Ash, J. S., Sittig, D. F., Poon, E. G., Guappone, K., Campbell, E., & Dykstra, R. H. (2007). The Extent and Importance of Unintended Consequences Related to Computerized Provider Order Entry. *Journal of the American Medical Informatics Association*, 14(4), 415–423. <https://doi.org/10.1197/jamia.M2373>
- Assis, D., & Souza, B. D. E. (2014). Um Panorama Do Uso De openEHR.
- Atalag, K., Yang, H. Y., Tempero, E., & Warren, J. (2011). Model Driven Development of Clinical Information Systems using openEHR. In *Studies in Health Technology and Informatics* (Vol. 169, pp. 849–853). <https://doi.org/10.3233/978-1-60750-806-9-849>
- Atalag, K., Yang, H. Y., & Warren, J. (2012). Assessment of Software Maintainability of openEHR Based Health Information Systems – A Case Study in Endoscopy. *Electronic Journal of Health Informatic*, 7(1), 1–10. Retrieved from <http://www.ejhi.net/ojs/index.php/ejhi/article/view/156>
- Beale, T. (2001). Archetypes: Constraint-based Domain Models for Future- proof Information Systems. *OOPSLA 2002 Workshop on Behavioural Semantics*, (21), 1–69. <https://doi.org/10.1.1.147.8835>
- Bizagi. (2014). *Bizagi BPM Suite. Topic relevant selected content from the* Retrieved from http://download.bizagi.com/docs/suite/1050/es/SuiteHelpEsp.pdf%5Cnhttp://samples.sainsburysebooks.co.uk/9781743044407_sample_137725.pdf#page=45
- BonitaSoft. (2014). Processes | Bonita Documentation. Retrieved from <http://documentation.bonitasoft.com/processes#installanapp>
- Bowens, F. M., Frye, P. A., & Jones, W. A. (2010). Health information technology: integration of clinical workflow into meaningful use of electronic health records. *Perspectives in Health Information Management / AHIMA, American Health Information Management Association*, 7, 1d. <https://doi.org/10.1586/erd.10.82>
- Buck, J., Garde, S., Kohl, C. D., & Knaup-Gregori, P. (2009). Towards a comprehensive electronic patient record to support an innovative individual care concept for premature infants using the openEHR approach. *International Journal of Medical Informatics*, 78(8), 521–531.

<https://doi.org/10.1016/j.ijmedinf.2009.03.001>

Combi, C., Gozzi, M., Oliboni, B., Juarez, J. M., & Marin, R. (2009). Temporal similarity measures for querying clinical workflows. *Artificial Intelligence in Medicine*, 46(1), 37–54.

<https://doi.org/10.1016/j.artmed.2008.07.013>

Conboy, K. (2009). Agility from first principles: Reconstructing the concept of agility in information systems development. *Information Systems Research*, 20(3), 329–354.

<https://doi.org/10.1287/isre.1090.0236>

Connolly, T., & Begg, C. (2005). *Database Systems*.

Correia, T. (2011). 05 - New Public Management in the Portuguese health sector: a comprehensive reading. *Sociologia On Line: Revista Da Associação Portuguesa de Sociologia*.

Davenport, T. H., & Beers, M. C. (1995). Managing Information about Processes. *Journal of Management Information Systems*, 12(1), 57–80. <https://doi.org/10.1080/07421222.1995.11518070>

Delgado, A., Calegari, D., Milanese, P., Falcon, R., & García, E. (2015). A systematic approach for evaluating BPM systems: Case studies on open source and proprietary tools. In *IFIP Advances in Information and Communication Technology* (Vol. 451, pp. 81–90). https://doi.org/10.1007/978-3-319-17837-0_8

Dingsøyr, T., Nerur, S., Balijepally, V., & Moe, N. B. (2012). A decade of agile methodologies: Towards explaining agile software development. *Journal of Systems and Software*, 85(6), 1213–1221.

<https://doi.org/10.1016/j.jss.2012.02.033>

Dumas, M., La Rosa, M., Mendling, J., & Reijers, H. A. (2013). *Fundamentals of Business Process Management. Fundamentals of Business Process Management*. <https://doi.org/10.1007/978-3-642-33143-5>

Elmasri, R., & Navathe, S. (2015). *Fundamentals of Database Systems 7th Edition* (7th ed.).

Entidade Reguladora da Saúde. (2011). Estudo sobre a organização e desempenho das Unidades Locais de Saúde: Relatório preliminar I, 1–39. Retrieved from https://www.ers.pt/uploads/document/file/5657/Plano_actividades_2015_ERS.pdf

Georgakopoulos, D., Hornick, M., & Sheth, A. (1995). An overview of workflow management: From process modeling to workflow automation infrastructure. *Distributed and Parallel Databases*, 3(2), 119–153. <https://doi.org/10.1007/BF01277643>

Gillham, B. (2000). Case study research methods. *Continuum*, 58. <https://doi.org/10.1007/s13398-014-0173-7.2>

Häyrynen, K., Saranto, K., & Nykänen, P. (2008). Definition, structure, content, use and impacts of

- electronic health records: A review of the research literature. *International Journal of Medical Informatics*. <https://doi.org/10.1016/j.ijmedinf.2007.09.001>
- Hevner, A. R., March, S. T., Park, J., & Ram, S. (2004). Design Science in Information Systems Research. *MIS Quarterly*, *28*(1), 75–105. <https://doi.org/10.2307/25148625>
- Hoerbst, A., & Ammenwerth, E. (2010). Electronic health records: A systematic review on quality requirements. *Methods of Information in Medicine*. <https://doi.org/10.3414/ME10-01-0038>
- Hulukua, D., & Popov, O. (2012). Root Cause Analysis of Session Management and Broken Authentication Vulnerabilities. *Internet Security (WorldCIS), 2012 World Congress On*, 82–86.
- International Organization For Standardization. (2005). Health informatics - Electronic health record - Definition, scope and context. *Definition of Electronic Health Record, ISO/TR 205*, 27. [https://doi.org/ISO/TR 20514:2005\(E\)](https://doi.org/ISO/TR 20514:2005(E))
- Kilsdonk, E., Peute, L. W. P., Knijnenburg, S. L., & Jaspers, M. W. M. (2011). Factors known to influence acceptance of clinical decision support systems. In *Studies in Health Technology and Informatics*. <https://doi.org/10.3233/978-1-60750-806-9-150>
- Lee, J., & Shartzter, A. (2005). Health IT and Workflow in Small Physicians ' Practices. *NIHCM Foundation*, (April).
- Luftman, J. (2003). Assessing IT business alignment. *Information Systems Management*, *20*(4), 9–15. <https://doi.org/10.1201/1078/43647.20.4.20030901/77287.2>
- Militello, L. G., Arbuckle, N. B., Saleem, J. J., Patterson, E., Flanagan, M., Haggstrom, D., & Doebbeling, B. N. (2014). Sources of variation in primary care clinical workflow: Implications for the design of cognitive support. *Health Informatics Journal*, *20*(1), 35–49. <https://doi.org/10.1177/1460458213476968>
- Ministério da Saúde. (2009). RSE – Registo de Saúde Electrónico.
- Mohammad, N., Munassar, A., & Govardhan, A. (2010). A Comparison Between Five Models Of Software Engineering. *International Journal of Computer Science Issues*, *7*(5), 94–101. <https://doi.org/10.1.1.402.9250>
- Niazkhani, Z., Pirnejad, H., Berg, M., & Aarts, J. (2009). The Impact of Computerized Provider Order Entry Systems on Inpatient Clinical Workflow: A Literature Review. *Journal of the American Medical Informatics Association*, *16*(4), 539–549. <https://doi.org/10.1197/jamia.M2419>
- Pagliari, C., Detmer, D., & Singleton, P. (2007). Potential of electronic personal health records. *BMJ*, *335*(7615), 330–333. <https://doi.org/10.1136/bmj.39279.482963.AD>
- Patterson, E. S., Gurses, A. P., Simmons, D., & Gibbons, M. C. (2014). Integrating Electronic Health

- Records into Clinical Workflow : An Application of Human Factors Modeling Methods to Ambulatory Care. *Proceedings of the International Symposium on Human Factors and Ergonomics in Health Care*, 3(1), 39 pages. <https://doi.org/10.6028/NIST.IR.7988>
- Pearce, C., & Bainbridge, M. (2014). A personally controlled electronic health record for Australia. *Journal of the American Medical Informatics Association*, 21(4), 707–713. <https://doi.org/10.1136/amiajnl-2013-002068>
- ProcessMaker. (2018). Wiki ProcessMaker. Retrieved from <https://wiki.processmaker.com/>
- Rodríguez, J. M., Barca, C. C., Puddu, P. E., Gialelis, J., Chondros, P., Karadimas, D., ... Verlinden, J. M. (2014). Archetype-based solution to tele-monitor patients with chronic diseases. In *IFMBE Proceedings* (Vol. 41, pp. 1278–1281). https://doi.org/10.1007/978-3-319-00846-2_316
- Rohner, P. (2012). Achieving impact with clinical process management in hospitals: an inspiring case. *Business Process Management Journal*, 18(4), 600–624. <https://doi.org/10.1108/14637151211253756>
- Ruffolo, M., Manna, M., Cozza, V., & Ursino, R. (2007). Semantic clinical process management. In *Proceedings - IEEE Symposium on Computer-Based Medical Systems* (pp. 518–523). <https://doi.org/10.1109/CBMS.2007.98>
- Schuler, T., Garde, S., Heard, S., & Beale, T. (2006). Towards automatically generating graphical user interfaces from openEHR archetypes. *Studies in Health Technology and Informatics*, 124(October 2015), 221–226. <https://doi.org/10.3233/978-1-58603-647-8-221>
- Silver, B. (2011). *BPMN Method and Style, 2nd Edition, with BPMN Implementer's Guide: A structured approach for business process modeling and implementation using BPMN 2.0*. Cody-Cassidy Press.
- Stohr, E. A., & Zhao, J. L. (2001). Workflow Automation: Overview and Research Issues. *Information Systems Frontiers*, 3(3), 281–296. <https://doi.org/10.1023/A:1011457324641>
- Taleb, M., Seffah, A., & Abran, A. (2010). Investigating Model-Driven Architecture for Web-based Interactive Systems. *Investigating Model-Driven Architecture for Web-Based Interactive Systems*, 1(6), 16.
- Vaishnavi, V., & Kuechler, B. (2008). Design Science Research in Information Systems. *Ais*, 45. <https://doi.org/10.1007/978-1-4419-5653-8>
- Van Der Aalst, W. M. P., La Rosa, M., & Santoro, F. M. (2016). Business process management: Don't forget to improve the process! *Business and Information Systems Engineering*, 58(1), 1–6. <https://doi.org/10.1007/s12599-015-0409-x>
- Veiga, M. V. (2015). *Processo Clínico Eletrónico*. Faculdade de Direito da Universidade de Coimbra.

- Weidenhaupt, K., & Lagor, C. (2008). Clinical Workflow Management And Decision System and Method. WfMC. (1999). Workflow Management Coalition Terminology {&} Glossary. *Management*, 39(3), 1–65. [https://doi.org/10.1016/S0019-0578\(00\)00014-8](https://doi.org/10.1016/S0019-0578(00)00014-8)
- Wieggers, K. E., & Beatty, J. (2013). *Software Requirements 3rd Ed. Software Requirements 3rd Ed.* Retrieved from http://www.amazon.com/s/ref=nb_sb_noss?url=search-alias%3Daps&field-keywords=9780735679665
- Yao, W., & Kumar, A. (2013). CONFlexFlow: Integrating Flexible clinical pathways into clinical decision support systems using context and rules. *Decision Support Systems*. <https://doi.org/10.1016/j.dss.2012.10.008>
- Yin, R. K. (2009). *Case Study Research: Design and Methods. Essential guide to qualitative methods in organizational research* (Vol. 5). <https://doi.org/10.1097/FCH.0b013e31822dda9e>
- Zheng, K., Ciemins, E. L., Curt, L., & Holly Lanham, M. J. (2015). *Examining the Relationship Between Health IT and Ambulatory Care Workflow Redesign*. Retrieved from <https://healthit.ahrq.gov/sites/default/files/docs/citation/examining-the-relationship-between-health-it-and-ambulatory-care-workflow-redesign-final-report.pdf>

Apêndices

```
function pmRestLogin($clientId, $clientSecret, $username, $password) {
    global $pmServer, $pmWorkspace;
    $postParams = array(
        'grant_type' => 'password',
        'scope'      => '', //set to 'view_process' if not changing the process
        'client_id'  => $clientId,
        'client_secret' => $clientSecret,
        'username'   => $username,
        'password'   => $password
    );

    $ch = curl_init("$pmServer/$pmWorkspace/oauth2/token");
    curl_setopt($ch, CURLOPT_TIMEOUT, 30);
    curl_setopt($ch, CURLOPT_POST, 1);
    curl_setopt($ch, CURLOPT_POSTFIELDS, $postParams);
    curl_setopt($ch, CURLOPT_RETURNTRANSFER, true);

    $oToken = json_decode(curl_exec($ch));
    $httpStatus = curl_getinfo($ch, CURLINFO_HTTP_CODE);
    curl_close($ch);

    if ($httpStatus != 200) {
        //print "Error in HTTP status code: $httpStatus\n";
        header('location:index.php?code='.$httpStatus);
        return null;
    }
    elseif (isset($oToken->error)) {
        print "Error logging into $pmServer:\n" .
            "Error:      {$oToken->error}\n" .
            "Description: {$oToken->error_description}\n";
    }
    else {
        //At this point $oToken->access_token can be used to call REST endpoints.

        //If planning to use the access_token later, either save the access_token
        //and refresh_token as cookies or save them to a file in a secure location.

        //If saving them as cookies:
        setcookie("access_token", $oToken->access_token, time() + 86400);
        setcookie("refresh_token", $oToken->refresh_token); //refresh token doesn't expire
        setcookie("client_id", $clientId);
        setcookie("client_secret", $clientSecret);
        setcookie("username", $username);

        //If saving to a file:
        //file_put_contents("/secure/location/oauthAccess.json", json_encode($tokenData));
    }
    //print $_COOKIE["access_token"];

    return $oToken;
}
```

Figura 40 - Função pmRestLogin

```

function pmRestRequest($method, $endpoint, $aVars = null) {
    global $pmServer;

    if (isset($_COOKIE['access_token']))
        $accessToken = $_COOKIE['access_token'];

    if (empty($accessToken)) { //if the access token has expired
        //To check if the PM login session has expired: !isset($_COOKIE['PHPSESSID'])
        header("Location: index.php"); //change to match your login method
        die();
    }

    //add beginning / to endpoint if it doesn't exist:
    if (!empty($endpoint) and $endpoint[0] != "/")
        $endpoint = "/" . $endpoint;

    $ch = curl_init($pmServer . $endpoint);
    curl_setopt($ch, CURLOPT_HTTPHEADER, array("Authorization: Bearer " . $accessToken));
    curl_setopt($ch, CURLOPT_TIMEOUT, 30);
    curl_setopt($ch, CURLOPT_RETURNTRANSFER, true);
    $method = strtoupper($method);

    switch ($method) {
        case "GET":
            curl_setopt($ch, CURLOPT_CUSTOMREQUEST, "GET");
            break;
        case "DELETE":
            curl_setopt($ch, CURLOPT_CUSTOMREQUEST, "DELETE");
            break;
        case "PUT":
            curl_setopt($ch, CURLOPT_CUSTOMREQUEST, "PUT");
            if($aVars != null){
                curl_setopt($ch, CURLOPT_POSTFIELDS, http_build_query($aVars));
            }
            break;
        case "POST":
            curl_setopt($ch, CURLOPT_POST, 1);
            if($aVars != null){
                curl_setopt($ch, CURLOPT_POSTFIELDS, http_build_query($aVars));
            }
            break;
        default:
            throw new Exception("Error: Invalid HTTP method '$method' $endpoint");
            return null;
    }

    $oRet = curl_exec($ch);
    $status = curl_getinfo($ch, CURLINFO_HTTP_CODE);
    curl_close($ch);

    $oRet = json_decode($oRet, true);

    if ($status == 401) { //if session has expired or bad login:
        header("Location: index.php"); //change to match your login method
        die();
    }

    elseif ($status != 200 and $status != 201) { //if error
        if ($oRet and isset($oRet)) {
            print "Error in $pmServer:\nCode: {"$oRet['error']['code']}\n" .
                "Message: {"$oRet['error']['message']}\n";
        }
        else {
            print "Error: HTTP status code:" . $status . "\n";
        }
    }

    return $oRet;
}

```

Figura 41 - Função pmRestRequest

```

function fillLista($list, $title)
{
    if (count($list) > 0) {
        >>
        <button type="button" class="collapsible"><?php echo count($list); ?></button>
        <div class="content">
            <table class="table table-hover table-bordered">
                <thead>
                    <tr class="bg-primary">
                        <th>#</th>
                        <th>Número do Caso</th>
                        <th>Processo</th>
                        <th>Tarefa</th>
                        <th>Enviado por</th>
                        <th>Estado</th>
                    </tr>
                </thead>
                <tbody>
                    <?php
                    $i = 0;
                    foreach ($list as $case) {
                        $i++;
                        echo <tr><td>" . $i . "</td>
                            <td>" . $case['app_number'] . "</td>
                            <td>" . $case['app_pro_title'] . "</td>
                            <td>" . $case['app_tas_title'] . "</td>
                            <td>" . $case['app_del_previous user'] . "</td>
                            <td>" . $case['app_status'] . "</td>
                            <td><a href='dynaform.php?proj=' . $case['pro_uid'] . "&task=" . $case['tas_uid'] . "&app=" . $case['app_uid'] . "' class='btn btn-default btn-xs'>Abrir</a></td></tr>";
                    }
                </tbody>
            </table>
        </div>
    </div>
    <?php
    }
}

```

Figura 42 - Função fillLists

```

if(isset($_POST['processSelection'])){
    $processSelected = $_POST['processSelected'];
    $arr = explode('/', $processSelected);
    $processId = $arr[0];
    $taskId = $arr[1];
    $aVars = array(
        'pro_uid' => $processId,
        'tas_uid' => $taskId
    );
    $oRet = pmRestRequest("POST", "/api/1.0/workflow/cases/", $aVars);
    header('Location: dynaform.php?proj=' . $processId . '&task=' . $taskId . '&app=' . $oRet['app_uid']);
}

```

Figura 43 - Código: Iniciar Caso

```

if(isset($_GET['doc'])){
    $doc = pmRestRequest('GET', '/api/1.0/workflow/cases/' . $_GET['app'] . '/output-documents', null);
    $toDownload = pmRestRequest('GET', "/api/1.0/workflow/cases/" . $_GET['app'] . "/output-document/" . $doc[0]['app_doc_uid'], null);
    var_dump($toDownload);
    if(isset($toDownload)){
        $link = "http://127.0.0.1/sysworkflow/en/neoclassic/" . $toDownload['app_doc_link'];
        //Download binary PDF file:
        $handle = fopen($link, "rb") or die("Error: Unable to open file:\n$link");
        $contents = stream_get_contents($handle) or die("Error: Unable to get file:\n$link");
        fclose($handle);
        //save file to a app_doc_filename on local harddrive
        $file = file_put_contents('D:/Programas/' . $_GET['caseNumber'] . $toDownload['app_doc_filename'], $contents);
    }

    if($file){
        $bool = 1;
    }else{
        $bool = 2;
    }
    header('Location: list.php?docBol='.$bool);
}

```

Figura 44 - Código: Descarregar informação do workflow

```

if(isset($_GET['task'])){
    $steps = pmRestRequest('GET', '/api/1.0/workflow/project/' . $_GET['proj'] . '/activity/' . $_GET['task'] . '/steps', null);
    $dynaform = pmRestRequest('GET', '/api/1.0/workflow/project/' . $_GET['proj'] . '/dynaform/' . $steps[0]['step_uid_obj']);
    $variables = pmRestRequest('GET', '/api/1.0/workflow/cases/' . $_GET['app'] . '/variables', null);

    $content = json_decode($dynaform['dyn_content'], true);
    $routeVariables = array();

    foreach ($content['items'][0]['items'] as $field) {
        for ($i = 0; $i <= count($field) - 1; $i++) {
            if($field[$i]['var_name'] == "linkToForm"){
                $linkToForm = $field[$i]['defaultValue'];
            }else{
                array_push($routeVariables, $field[$i]['var_name']);
            }
        }
    }
    $serializedArray = htmlentities(serialize($routeVariables));
}

```

Figura 45 - Código: Dynaform

```

if(is null($_COOKIE['access_token']))
    header('Location: index.php');

if(isset($_GET['app'])){
    $app = $_GET['app'];
    $variables = $_GET['variables'];
    $variables = unserialize($variables);
    $currentUser = $_GET['user'];
}

$data = array();
if(isset($_POST)){
    foreach($_POST as $key => $value){
        if(in_array($key, $variables)){
            $data[$key] = $value;
        }
    }
}

if(count($data) > 0){
    pmRestRequest('PUT', '/api/1.0/workflow/cases/' . $app . '/variable', $data);
}else{
    pmRestRequest('PUT', '/api/1.0/workflow/cases/' . $app . '/variable', null);
}

pmRestRequest('PUT', '/api/1.0/workflow/cases/' . $app . '/route-case', null);
$caseInfo = pmRestRequest('GET', '/api/1.0/workflow/cases/' . $app);
var_dump($caseInfo);
// If the next task is assigned to the same user, go for the next dynaform
$control = false;
foreach($caseInfo['current_task'] as $nextUser){
    if($_GET['user'] == $nextUser['usr_uid']){
        $control = true;
    }
}

if($control == true){
    header("Location: dynaform.php?proj=" . $caseInfo['pro_uid'] . "&task=" . $caseInfo['current_task'][0]['tas_uid'] . "&app=" . $caseInfo['app_uid'] );
}else{
    header("Location: dynaform.php?id=1");
}

```

Figura 46 - Código Route_Case

```

if($caseInfo['app_status'] == 'COMPLETED'){
    $docs = pmRestRequest('GET', '/api/1.0/workflow/project/' . $caseInfo['pro_uid'] . '/output-documents', null);
    foreach($docs as $output){
        if($output['out_doc_filename'] == 'informacaoCaso'){
            $out = pmRestRequest("POST", '/api/1.0/workflow/cases/' . $app . '/2/output-document/' . $output['out_doc_uid'], null);
            var_dump($out);
        }
    }
}
}

```

Figura 47 - Código: Gerar documento com informação do workflow

```

Trigger
1 $numUtente = @@numUtente;
2
3 $queryUtente = "SELECT * FROM utente WHERE numUtente='$numUtente'";
4 $countUtente = executeQuery($queryUtente, "2809448245b8069336d1452061541877");
5
6 if(count($countUtente) > 0){
7     @@existeUtente = 'Sim';
8     @@utente = $countUtente;
9     $query = "SELECT * FROM exame e, tipoexame te, utente u WHERE idUtente = (Select id From Utente
10 Where numUtente = '$numUtente') AND e.idTipoExame = te.idtipoExame AND e.idUtente = u.id";
11 $result = executeQuery($query, "2809448245b8069336d1452061541877");
12 if(count($result) > 0){
13     @@exame = $result;
14     @@existeExame = 'Sim';
15 }
16 else{
17     @@existeExame = 'NÃO';
18 }
19 }else{
20     @@existeUtente = 'NÃO';
21 }
22

```

Figura 48 - Trigger "Verificar utente e exames"

```

Trigger
1 $caseId = @@APPLICATION;
2 $result = executeQuery("SELECT APP_NUMBER, APP_CREATE_DATE FROM APPLICATION WHERE APP_UID='$caseId'");
3 @%caseNumber = $result[1]['APP_NUMBER'];
4 @%caseBeginDate = $result[1]['APP_CREATE_DATE'];
5
6 $processId = @@PROCESS;
7 $process = executeQuery("SELECT PRO_TITLE FROM PROCESS WHERE PRO_UID='$processId'");
8 @@ProcessName = $process[1]['PRO_TITLE'];
9
10
11

```

Figura 49 - Trigger "Preencher variáveis para documento"

```

Trigger
1 if(@@isExameSuficiente == 'Sim' || @@autorizarExame == 'NÃO' || @@aceitarExame == 'NÃO' ||
2 (@@isExameSuficiente=='NÃO' && @@autorizarExame == 'Sim' && @@aceitarExame=='Sim')){
3 $exameSuficiente = @@isExameSuficiente;
4 $existeExame = @@existeExame;
5 $submeterExame = @@autorizarExame;
6 $aceitarExame = @@aceitarExame;
7 $justificarRecusa = @@justificarRecusa;
8 $numUtente = @@numUtente;
9
10 $query = "INSERT INTO workflow (existeExame, exameSuficiente, submeterExame, aceitarExame,
11 justificarRecusa, idTipoWorkflow, numUtente) VALUES ('$existeExame', '$exameSuficiente',
12 '$submeterExame', '$aceitarExame', '$justificarRecusa', '1', '$numUtente')";

```

Figura 50 - Trigger "Guardar workflow"

```
Trigger
1 $aUser = userInfo(@@USER_LOGGED);
2 @@name = $aUser['firstname'].' '.$aUser['lastname'];
3 @@emailHP = $aUser['mail'];
```

Figura 51 - Trigger "Dados Profissional de Saúde"

```
Trigger
1 $aUser = userInfo(@@USER_LOGGED);
2 @@fullTechnicianName = $aUser['firstname'].' '.$aUser['lastname'];
```

Figura 52 - Trigger "Dados Técnico"

Email Event Properties

To*: @@emailHP

Subject: Justificação de Recusa de Exame @@

Content*:
Upload File @# B I U advanced.font advanced.font
Dr./Dra. @=name,
O seu pedido de exame foi recusado devido às seguintes razões:
@=justificarRecusa.
Cumprimentos,
@=fullTechnicianName

advanced.path: p Words:17

Figura 53 - Evento "Envio de email de recusa"

Pedido de Exame TAC

Preencha os seguintes dados do Utente

Nome	Idade
<input type="text"/>	<input type="text"/>
Sexo	Alergias
<input type="text" value="Selecione um valor"/>	<input type="text"/>
Peso	Possui Pacemaker?
<input type="text"/>	<input type="checkbox"/>
Motivo para o pedido de Exame	
<input type="text"/>	
Submeter Pedido de Exame	Justificação de Recusa
<input type="radio"/> Submeter	<input type="text" value="Preencher caso tenha escolhido a opção Não Submeter"/>
<input type="radio"/> Não Submeter	

Figura 54 - Formulário "Recolher variáveis" para Utente não existente no sistema



Figura 55 - Mensagem de reencaminhamento do Caso

Pedido de Exame TAC

Utente: Tiago Ribeiro

TAC - 30/09/2018
TAC - 17/10/2018
Eletrocardiograma - 15/11/2018

Exames são suficientes

Exames não são suficientes

Figura 56 - Formulário "Exames Suficientes" após submissão do workflow exemplo