



# Sistema de Recomendação de Recursos Humanos e Gestão de informação em Projetos de I&D

**TIAGO MANUEL TEIXEIRA CARDOSO**

Outubro de 2020

# **Sistema de Recomendação de Recursos Humanos e Gestão de informação em Projetos de I&D**

**Tiago Manuel Teixeira Cardoso**

**Dissertação para obtenção do Grau de Mestre em  
Engenharia Informática, Área de Especialização em  
Engenharia de Software**

**Orientador: Maria Goreti Carvalho Marreiros**

**Co-orientador: Luís Manuel da Silva Conceição**



# Resumo

A gestão de projetos é uma tarefa de grande complexidade para muitas organizações, pois para lá de um grande número de projetos que possam existir numa organização cada um destes tem vários aspetos que necessitam de ser abordados. Entre estes aspetos existe a gestão dos recursos humanos afetos a cada projeto, a gestão das diferentes variáveis financeiras, gestão de atividades e objetivos envolvidos na realização do projeto em questão e a gestão da informação gerada durante a sua execução.

O GECAD, como grupo de investigação, também se depara com este problema, tornando-se assim imprescindível a utilização de uma ferramenta que permita controlar e planear a gestão destes diferentes aspetos, especialmente de modo a otimizar a execução dos projetos quer em termos científicos, quer em termos financeiros. Para o efeito existe um portal que permite a gestão de projetos ao nível dos recursos humanos afetos a cada projeto, assim como dos recursos financeiros disponíveis. No entanto, ainda existem alguns aspetos que o portal existente não consegue tratar, exemplo disso é a recomendação de recursos humanos para novos projetos e a disponibilização/apresentação de informação de forma agregada (*dashboards*).

Neste documento é abordado o desenvolvimento de um Sistema de Recomendação que pretende facilitar a alocação de investigadores às várias candidaturas de projetos elaboradas pelo GECAD tendo por base o perfil de cada investigador e as características das tarefas envolvidas em cada uma das candidaturas. Para além disso, foi desenvolvido um componente que permite agrupar informação relativa aos projetos em execução e já executados, afetação de investigadores e indicadores dos vários projetos de modo a disponibilizá-la de uma forma precisa e facilmente compreensível aos utilizadores.

Estes novos componentes foram integrados com a solução original, implicando várias alterações no domínio e funcionalidades desta de modo a permitir esta integração. Após a criação desta nova solução, esta foi avaliada através de um inquérito de satisfação e mais especificamente no caso do Sistema de Recomendação foram realizados diferentes testes estatísticos. Através destes diferentes métodos de avaliação, determinou-se que de um ponto de vista da solução como um todo, esta foi capaz de ir de encontro às necessidades do cliente e apresentar resultados satisfatórios.

**Palavras-Chave:** Sistema de Recomendação, *Dashboards*, Gestão de Recursos Humanos

# Abstract

Project management is a task of great complexity for many organizations, because beyond the large number of projects that may exist in an organization, each of these has several aspects that need to be addressed. Among these aspects there is the management of human resources allocated to each project, the management of the different financial variables, the management of activities and objectives involved in carrying out the project in question and the management of the information generated during its execution.

GECAD, as a research group, is also faced with this problem, making it essential to use a tool that allows one to control and plan the management of these different aspects, especially in order to optimize the execution of projects both in scientific and financial terms. For this purpose, there is a portal that allows the management of projects in terms of the human resources allocated to each project, as well as the financial resources available. However, there are still some aspects that the existing portal is unable to address, an example of which is the recommendation of human resources for new projects and the availability/presentation of information in an aggregated form (dashboards).

This document discusses the development of a Recommendation System that aims to facilitate the allocation of researchers to the various project applications prepared by GECAD based on the profile of each researcher and the characteristics of the tasks involved in each of the applications. In addition, a component was developed that allows the grouping of information relating to projects in execution and already executed, the allocation of researchers and indicators of the various projects in order to make it available in a precise and easily understandable way.

These new components were integrated with the original solution, implying several changes in its domain and functionalities to allow this integration. After the creation of this new solution, it was evaluated through a satisfaction survey and in the case of the Recommendation System, different statistical tests were carried out. Through these different evaluation methods, it was determined that from the point of view of the solution as a whole, it was able to meet the needs of the client and present satisfactory results.

**Keywords:** Recommendation System, Dashboards, Human Resource Management



# Agradecimentos

Gostaria de agradecer ao GECAD por me ter dado acesso às suas instalações, informação e investigadores durante o processo de desenvolvimento deste projeto. Mais especificamente gostaria de agradecer aos meus orientadores Maria Goreti Carvalho Marreiros e Luís Manuel da Silva Conceição, por me terem ajudado e encaminhado durante o desenvolvimento deste projeto.

Finalmente, agradeço à minha família e amigos, que me apoiaram ao longo do desenvolvimento deste projeto e durante todo o meu percurso académico.



# Índice

<b>1</b>	<b>Introdução.....</b>	<b>1</b>
1.1	Motivação .....	1
1.2	Enquadramento.....	2
1.3	Problema.....	3
1.4	Objetivos.....	4
1.5	Estrutura da dissertação.....	4
<b>2</b>	<b>Contexto e Estado da Arte.....</b>	<b>7</b>
2.1	GesProj.....	7
2.2	Abordagens Existentes.....	16
2.2.1	Recomendações baseadas em conteúdo.....	17
2.2.2	Recomendações colaborativas .....	17
2.2.3	Abordagens híbridas .....	18
2.2.4	Medidas de Semelhança .....	20
2.2.5	Métricas de Avaliação .....	21
2.2.6	Recomendação de Recursos Humanos para Equipas .....	22
2.3	Soluções Existentes.....	24
2.3.1	ReSySTER.....	24
2.3.2	Social Web Application for Team Recommendation (SWAT) .....	24
2.3.3	Opportunity Team Builder (OTB) .....	25
2.4	Avaliar Soluções e Abordagens Existentes .....	26
2.4.1	AHP.....	26
2.4.2	Avaliação de Abordagens.....	27
2.4.3	Avaliação de Soluções.....	33
2.5	Tecnologias.....	34
2.6	Análise de Valor .....	35
2.6.1	Modelo de Desenvolvimento de Novos Conceitos .....	36
2.6.2	Valor Percebido.....	38
2.6.3	Proposta de Valor .....	39
2.6.4	Quality function deployment (QFD).....	41
2.6.5	Modelo Canvas.....	44

<b>3</b>	<b>Design .....</b>	<b>47</b>
<b>3.1</b>	<b>Requisitos.....</b>	<b>47</b>
3.1.1	Requisitos Funcionais.....	47
3.1.2	Requisitos Não Funcionais .....	50
<b>3.2</b>	<b>Arquitetura.....</b>	<b>51</b>
<b>3.3</b>	<b>Conceitos do Domínio .....</b>	<b>53</b>
<b>3.4</b>	<b>Base de Dados .....</b>	<b>56</b>
<b>3.5</b>	<b>Processos .....</b>	<b>59</b>
3.5.1	Recomendação de recursos humanos .....	59
3.5.2	Consulta de percentagens de afetação dos investigadores aos projetos .....	60
3.5.3	Consulta da <i>timeline</i> de um projeto.....	61
3.5.4	Consulta de indicadores por projeto .....	62
<b>4</b>	<b>Desenvolvimento .....</b>	<b>65</b>
<b>4.1</b>	<b>SQLServerDb.....</b>	<b>65</b>
<b>4.2</b>	<b>PortalRHGecadBackEnd .....</b>	<b>65</b>
4.2.1	Consulta de uma Área com base no seu id .....	66
4.2.2	Testes .....	69
<b>4.3</b>	<b>PortalRHGecadQuerier.....</b>	<b>71</b>
4.3.1	Implementação das Consultas .....	72
4.3.1.1	Consulta de percentagens de afetação dos investigadores aos projetos .....	72
4.3.1.2	Consulta da <i>timeline</i> de um projeto .....	74
4.3.1.3	Consulta de indicadores por projeto .....	75
4.3.2	DTOFactory .....	75
4.3.3	Testes .....	76
<b>4.4</b>	<b>PortalRHGecadRecommender .....</b>	<b>78</b>
4.4.1	Implementação das Abordagens .....	81
4.4.1.1	Abordagem baseada em conteúdo .....	81
4.4.1.2	Abordagem colaborativa (Microsoft.ML) .....	83
4.4.1.3	Abordagem colaborativa (KNN).....	85
4.4.1.4	Abordagens híbridas.....	87
4.4.1.5	Comparação .....	87
4.4.2	DTOFactory .....	88
4.4.3	Testes .....	89

<b>4.5</b>	<b>PortalRHGecadFrontEnd .....</b>	<b>90</b>
4.5.1	GesProj Revisto .....	90
4.5.2	Recomendações .....	93
4.5.3	<i>Dashboards</i> .....	94
4.5.3.1	Afetações .....	95
4.5.3.2	Indicadores .....	98
4.5.3.3	<i>Timeline</i> .....	99
4.5.4	Introduzir Área .....	100
<b>5</b>	<b><i>Avaliação</i> .....</b>	<b>103</b>
<b>5.1</b>	<b>Hipóteses .....</b>	<b>103</b>
<b>5.2</b>	<b>Metodologia de Avaliação .....</b>	<b>103</b>
<b>5.3</b>	<b>Testes Estatísticos .....</b>	<b>104</b>
<b>5.4</b>	<b>Sistema de Recomendação .....</b>	<b>105</b>
5.4.1	Cenários de Teste .....	105
5.4.2	Análise de Resultados .....	109
5.4.3	Conclusões .....	114
<b>5.5</b>	<b><i>Dashboards</i> e a Solução em Geral .....</b>	<b>115</b>
5.5.1	Inquérito de Satisfação .....	115
5.5.2	Análise de Resultados .....	118
5.5.3	Conclusões .....	119
<b>6</b>	<b><i>Conclusão</i> .....</b>	<b>121</b>
<b>6.1</b>	<b>Apreciação Final .....</b>	<b>121</b>
<b>6.2</b>	<b>Limitações e Trabalho futuro .....</b>	<b>123</b>
	<b><i>Bibliografia</i> .....</b>	<b>125</b>
	<b><i>Anexo I – Exemplos de folhas Excel</i> .....</b>	<b>129</b>
	Folha de afetação global .....	129
	Folha de afetação de um projeto .....	130
	<b><i>Anexo II – Resultados dos Cenários de Avaliação</i> .....</b>	<b>131</b>
	<b><i>Anexo III – Respostas ao Inquérito</i> .....</b>	<b>133</b>
	<b><i>Anexo IV – Scripts R</i> .....</b>	<b>135</b>

<b>Sistema de Recomendações .....</b>	<b>135</b>
<b><i>Dashboards</i> .....</b>	<b>141</b>

# Índice de Figuras

Figura 1 – Modelo de Domínio do GesProj .....	9
Figura 2 – Diagrama de Casos de Uso do GesProj.....	11
Figura 3 - Diagrama de componentes do GesProj .....	11
Figura 4 – Página inicial do portal .....	12
Figura 5 – Página de gestão de um projeto .....	13
Figura 6 – Página de gestão de indicadores.....	13
Figura 7 – Página de gestão de horas trabalháveis.....	14
Figura 8 – Página das horas trabalháveis de um ano.....	14
Figura 9 – Página de gestão de investigadores .....	15
Figura 10 - Categorias de sistemas de híbridos de recomendação.....	19
Figura 11 - Níveis de importância de comparações .....	27
Figura 12 – Valores IR.....	27
Figura 13 - Árvore Hierárquica .....	28
Figura 14 – Modelo NCD .....	36
Figura 15 – Processo de Inovação .....	38
Figura 16 – Canvas da Proposta de Valor.....	39
Figura 17 – Estrutura da Casa de Qualidade .....	42
Figura 18 – Símbolos do Teto.....	43
Figura 19 – Símbolos da Matriz.....	43
Figura 20 – Casa de Qualidade .....	44
Figura 21 – Modelo Canvas.....	45
Figura 22 – Diagrama de Casos de Uso .....	50
Figura 23 – Arquitetura 1 .....	51
Figura 24 – Arquitetura 2 .....	52

Figura 25 – Modelo de Domínio.....	56
Figura 26 – Modelo da Base de Dados.....	58
Figura 27 – Diagrama de sequência da recomendação de recursos humanos.....	59
Figura 28 – Diagrama de sequência da consulta de afetações aos projetos .....	60
Figura 29 – Diagrama de sequência da consulta da timeline de um projeto .....	61
Figura 30 – Diagrama de consulta de indicadores por projeto.....	62
Figura 31 – Estrutura de diretórios do PortalRHGecadBackEnd.....	66
Figura 32 – Classe Subject.....	67
Figura 33 – Método da classe SubjectRepository .....	67
Figura 34 – Método base de consulta com id.....	67
Figura 35 – Método da classe SubjectService.....	68
Figura 36 – Classe SubjectProfile .....	68
Figura 37 – Método da classe SubjectController .....	69
Figura 38 – Teste sobre a consulta de Área por id.....	70
Figura 39 – Testes de integração do back-end.....	70
Figura 40 – Resultado dos testes de integração do back-end .....	71
Figura 41 - Estrutura de diretórios do PortalRHGecadQuerier .....	71
Figura 42 – Método da classe QueryController .....	73
Figura 43 – Método de Afetações da classe QueryService .....	73
Figura 44 – Método da classe AffectationQuery .....	73
Figura 45 – Método de Timeline da classe QueryService .....	74
Figura 46 – DTO de Afetação num ano e mês.....	76
Figura 47 – DTOFactory de Afetação num ano e mês.....	76
Figura 48 – Porção de um teste unitário sobre a consulta de afetações.....	77
Figura 49 – Testes de integração do componente PortalRHGecadQuerier .....	78

Figura 50 – Resultado dos testes de integração do componente PortalRHGecadQuerier.....	78
Figura 51 - Estrutura de diretórios do PortalRHGecadRecommender .....	79
Figura 52 – Método de geração de recomendações do RecommenderService .....	80
Figura 53 – Método de recomendações baseadas em conteúdo.....	81
Figura 54 – Método de prioridade de investigadores.....	82
Figura 55 – Método de recomendações com ML .....	83
Figura 56 – Formatação de informação para utilização em ML.....	84
Figura 57 – Aplicação de ML .....	85
Figura 58 – Método de geração de informação com base em input.....	86
Figura 59 – Método de geração de recomendações híbridas.....	87
Figura 60 – Teste sobre recomendações .....	89
Figura 61 – Corpo de pedido HTTP POST .....	90
Figura 62 – Resultado dos testes Postman do componente de recomendações.....	90
Figura 63 – Nova página inicial do portal.....	91
Figura 64 – Página de gestão de áreas.....	91
Figura 65 – Página de gestão de indicadores por realizar .....	92
Figura 66 – Página de criação de indicadores realizados.....	92
Figura 67 – Página de gestão de milestones.....	93
Figura 68 – Página de recomendações de investigadores .....	93
Figura 69 – Página de recomendações com resultados.....	94
Figura 70 – Componente de seleção de dashboards .....	95
Figura 71 – Gráfico de afetações globais .....	96
Figura 72 – Gráfico de afetações filtrado.....	97
Figura 73 – Tabela de afetações .....	97
Figura 74 – Gráfico de indicadores .....	98

Figura 75 – Tabela de indicadores .....	99
Figura 76 – Gráfico da timeline de um projeto .....	99
Figura 77 – Método invocado pelo botão “Adicionar” .....	101
Figura 78 – Ações envolvidas na criação de uma nova área.....	101
Figura 79 – Box-plot das precisões das abordagens .....	111
Figura 80 – Box-plot do recall das abordagens.....	111
Figura 81 – Box-plot da f-measure das abordagens .....	112
Figura 82 – Gráfico da questão 1 .....	115
Figura 83 – Gráfico da questão 2 .....	116
Figura 84 – Gráfico da questão 3 .....	116
Figura 85 – Gráfico da questão 4 .....	117
Figura 86 – Gráfico da questão 5 .....	117
Figura 87 – Gráfico da questão 6 .....	118
Figura 88 – Gráfico da questão 7 .....	118

# Índice de Tabelas

Tabela 1 - Glossário .....	8
Tabela 2 - Estado de concretização do portal.....	15
Tabela 3 – Tabela Comparativa de Critérios .....	29
Tabela 4 - Tabela Comparativa de Critérios Normalizada.....	29
Tabela 5 - Tabela Comparativa de Cold-Start .....	30
Tabela 6 - Tabela Comparativa de Cold-Start Normalizada .....	31
Tabela 7 - Tabela Comparativa de Variabilidade .....	31
Tabela 8 - Tabela Comparativa de Variabilidade Normalizada .....	32
Tabela 9 - Tabela Comparativa de Performance.....	32
Tabela 10 - Tabela Comparativa de Performance Normalizada .....	33
Tabela 11 – Tabela de Avaliação de Soluções.....	34
Tabela 12 – Glossário de novos conceitos .....	55
Tabela 13 – Projetos para Avaliação .....	106
Tabela 14 – Investigadores para Avaliação.....	107
Tabela 15 – Tabela de cenários.....	108
Tabela 16 – Tabela de resultados para análise .....	110
Tabela 17 – Tabela de F-Measure para o teste de Friedman.....	112
Tabela 18 – Tabela de comparação com base em p-value .....	114
Tabela 19 – Tabela de kappa.....	119



# Acrónimos

<b>Acrónimos</b>	<b>Significado</b>
<b>AHP</b>	<i>Analitic Hierarchy Process</i>
<b>ATS</b>	<i>Applicant Tracking System</i>
<b>DTO</b>	<i>Data Transfer Object</i>
<b>FFE</b>	<i>Fuzzy Front End</i>
<b>GECAD</b>	Grupo de Investigação em Engenharia e Computação Inteligente para a Inovação e o Desenvolvimento
<b>IC</b>	Índice de Consistência
<b>ISEP</b>	Instituto Superior de Engenharia do Porto
<b>KNN</b>	<i>K-Nearest Neighbors Algorithm</i>
<b>MAE</b>	<i>Mean Absolute Error</i>
<b>MEI</b>	Mestrado em Engenharia Informática
<b>ML</b>	<i>Machine Learning</i>
<b>NCD</b>	<i>New Concept Development</i>
<b>OTB</b>	<i>Opportunity Team Builder</i>
<b>P-J</b>	<i>Person-job</i>
<b>P-T</b>	<i>Person-team</i>
<b>PESTI</b>	Projeto/Estágio
<b>QFD</b>	<i>Quality function deployment</i>
<b>RC</b>	Razão de Consistência
<b>RMSE</b>	<i>Root Mean Squared Error</i>
<b>SWAT</b>	<i>Social Web Application for Team Recommendation</i>
<b>TMDEI</b>	Tese/Dissertação/Estágio
<b>WP</b>	<i>Work Package</i>



# 1 Introdução

Neste capítulo é realizada uma introdução ao projeto descrito nesta dissertação, de modo a contextualizar o leitor. Primeiro é apresentada a motivação para a realização do projeto, seguida pelo enquadramento do mesmo, e finalmente uma análise ao problema apresentado, em conjunto com os objetivos do projeto. No final é apresentada a estrutura da dissertação, para mais uma vez auxiliar o leitor.

## 1.1 Motivação

A motivação para a realização deste projeto veio de tanto uma necessidade relacionada com o problema da gestão de projetos e de um interesse pessoal.

Do ponto de vista da gestão de projetos, esta é uma tarefa de grande complexidade para muitas organizações, pois implica lidar não só com os projetos em si, mas também com a gestão dos recursos humanos, aspetos financeiros, atividades e objetivos envolvidos na realização dos vários projetos. Estes diferentes aspetos também dão origem a diferentes tipos de informação que a organização necessita de manter. Daí ser natural uma organização sentir a necessidade de procurar métodos que lhe permita gerir estas diferentes facetas da gestão de projetos.

Quanto ao interesse pessoal, este surgiu durante a realização da unidade curricular de Projeto/Estágio (PESTI), onde o objetivo principal do projeto era desenvolver um algoritmo capaz de alocar alunos às várias optativas com base em diferentes critérios. Através deste projeto foi possível aplicar conceitos aprendidos durante a minha Licenciatura e desenvolver um interesse em algoritmos que facilitem ou ajudem a resolver um problema num contexto real.

Este interesse levou à escolha deste tema para o projeto da unidade curricular de Tese/Dissertação/Estágio (TMDEI), pois mais uma vez teria a oportunidade de desenvolver um algoritmo que permite esclarecer necessidades reais e adquirir assim mais experiência neste contexto. A escolha também foi influenciada pela possibilidade de voltar a trabalhar com a Professora Goreti que foi a minha orientadora durante o desenvolvimento do projeto de PESTI.

## 1.2 Enquadramento

Esta dissertação destina-se a documentar o desenvolvimento do projeto realizado no âmbito de TMDEI, do Mestrado em Engenharia Informática (MEI) no ramo de Engenharia de Software, do departamento de Engenharia Informática do Instituto Superior de Engenharia do Porto (ISEP) no ano letivo de 2019/2020.

O projeto realizou-se maioritariamente nas instalações do ISEP, mais especificamente no centro de investigação Grupo de Investigação em Engenharia e Computação Inteligente para a Inovação e o Desenvolvimento (GECAD). Este foi sugerido com base na ideia de introduzir melhorias num projeto desenvolvido no âmbito de PESTI, no GECAD durante o ano letivo de 2018/2019 (Couto 2019). Este projeto tinha como objetivo criar uma plataforma capaz de facilitar as diferentes tarefas relacionadas com a gestão de projetos. Esta plataforma será referida como GesProj ao longo deste documento.

Pretende-se assim facilitar a gestão de projetos, visto que é um processo moroso e de grande complexidade. Para exemplificar considere-se o caso da criação de uma equipa para uma candidatura de projeto, onde é necessário primeiro conhecer o perfil e capacidades dos vários investigadores existentes, as tarefas envolvidas na conceção do projeto e ainda as afetações dos investigadores a projetos que já estejam em execução.

Enquanto que uma candidatura apresenta um nível considerável de complexidade, é necessário mencionar que num ano o GECAD produz mais de 50 candidaturas, para as quais tem mais de 20 investigadores que podem ser alocados. Ao mesmo tempo existem mais de 30 projetos em execução que já têm investigadores alocados, limitando assim a sua disponibilidade para serem considerados para equipas de candidaturas.

Sendo assim, é revelada uma necessidade, ou oportunidade para acrescentar ao projeto existente algo que facilite estes aspetos, tal como é proposto nesta dissertação. Mais especificamente existe uma necessidade de um meio que facilite a escolha de investigadores quando é necessário decidir quais podem ser afetos a uma candidatura. Em conjunto com uma funcionalidade que permita consultar a informação gerada durante o processo de gestão de projetos.

### 1.3 Problema

O GesProj tem como finalidade a gestão de projetos, sendo esta uma tarefa complexa pois para cada projeto é necessário considerar a afetação de recursos humanos a cada projeto, realizar a gestão dos recursos financeiros disponíveis, e controlar os seus indicadores de realização atividades. A sua complexidade também deriva do facto do GECAD ter um número de projetos de investigação a decorrer, como por exemplo no período de 2018-2019 o GECAD executou 33 projetos de investigação, a que corresponde um financiamento de 7.5 milhões de euros.

Estes projetos são de diferentes tipologias, como por exemplo H2020, P2020, FCT, ou Erasmus, e obedecem a regras de financiamento muito específicas. Em vários destes é necessária a afetação de parte dos salários dos docentes e investigadores, o que implica que aquando da preparação de uma nova candidatura, se tenha em conta, para além do perfil dos possíveis membros da equipa, a sua disponibilidade.

No GesProj é possível alocar investigadores a projetos, contudo é necessário ter conhecimento prévio de quais são os investigadores a alocar ao projeto. Esta é uma tarefa que atualmente é feita manualmente e onde, para além da disponibilidade do investigador, são também consideradas as suas competências. Esta limitação, abre a possibilidade de incorporar no sistema a desenvolver uma funcionalidade capaz de sugerir investigadores, reduzindo assim parte do esforço colocado no processo de alocação de pessoas a projetos.

Uma outra limitação do GesProj é não disponibilizar informação agregada aos utilizadores, como por exemplo informação relativa ao estado de realização dos projetos ou afetação dos investigadores. Atualmente esta tarefa é realizada recorrendo a folhas Excel, que apesar de ser uma ferramenta que permite tratar e estruturar informação, é difícil de gerir tendo em conta a quantidade de informação presente neste contexto.

Como foi mencionado no Enquadramento num ano o GECAD chega a ter de tratar de mais de 50 candidaturas, mais de 30 projetos em execução e tem que gerir a afetação de mais de 20 investigadores por estes. Com o recurso a folhas de Excel a informação é agregada e estruturada de uma forma semelhante ao que é exemplificado no Anexo I, através das duas das folhas Excel apresentadas no anexo, com a informação sensível relativa aos investigadores e projetos alterada. Sendo assim cria-se uma necessidade de uma funcionalidade capaz de tratar e agregar a informação atualmente contida nestas folhas de Excel, para a depois disponibilizar aos diferentes utilizadores de uma forma simples, e intuitiva de utilização e interpretação.

Finalmente, há que ter em conta o facto de o GesProj ainda não estar totalmente funcional ou preparado para a resolução destes problemas, logo este necessita de uma revisão e implementação de alterações. Sendo estas baseadas na necessidade evolutiva do projeto abordado por esta dissertação, ou simplesmente melhorias nas funcionalidades já existentes.

## 1.4 Objetivos

Com base no portal existente e as necessidades apresentadas pelo GECAD, o objetivo geral do projeto é acrescentar ao portal a capacidade de fazer recomendações de recursos humanos para projetos e disponibilizar métodos de gestão de informação.

De modo a atingir este objetivo geral, é necessário atingir os seguintes objetivos específicos:

1. Análise e interpretação do problema, nomeadamente análise da plataforma do GesProj e identificação dos requisitos necessários;
2. Investigação e respetiva realização do estado da arte na área dos sistemas de recomendação;
3. Avaliação da informação adquirida durante a investigação do estado da arte;
4. Análise e modelação de componentes de recomendação e consulta de informação agregada que irão integrar com a plataforma GesProj;
5. Desenvolvimento dos componentes de recomendação e consulta de informação agregada;
6. Integração dos novos componentes com o GesProj;
7. Testes dos componentes desenvolvidos e avaliação dos resultados obtidos.

## 1.5 Estrutura da dissertação

Esta dissertação encontra-se dividida nos seguintes capítulos:

1. Introdução: Neste capítulo é realizado um enquadramento relativamente ao projeto a ser desenvolvido. É feita uma apresentação do problema a ser tratado nesta dissertação. São ainda expostos os objetivos e a estrutura da dissertação.
2. Contexto e Estado da Arte: Neste capítulo é realizada uma análise mais detalhada do ambiente onde o problema se integra. É apresentado o GesProj, em conjunto e um estudo relativo ao estado da arte do sistema de recomendação a ser desenvolvido. Também é realizado um breve estudo das tecnologias a serem utilizadas, uma avaliação

das soluções e abordagens existentes e finalmente é realizada a análise de valor da solução.

3. Design: Neste capítulo, são apresentados os requisitos do problema, é determinada a arquitetura da solução, e é realizada uma análise do domínio e de alguns dos processos da solução.
4. Implementação: Neste capítulo é apresentado o processo de implementação dos componentes e funcionalidades propostas no capítulo de Design.
5. Avaliação: Neste capítulo é realizada a avaliação da solução desenvolvida. Sendo definidas as hipóteses, metodologias de avaliação e testes estatísticos que irão ser utilizados no processo de avaliação. Finalmente é realizada a avaliação da solução desenvolvida.
6. Conclusão: Neste capítulo são apresentadas as conclusões atingidas nesta dissertação. Primeiro é realizada uma apreciação do trabalho desenvolvido, seguido da apresentação das suas limitações e trabalho futuro.



## **2 Contexto e Estado da Arte**

Neste capítulo, será realizada uma análise mais detalhada do ambiente onde o problema a ser tratado se integra. Para tal será inicialmente apresentado o GesProj para perceber o seu funcionamento atual, seguido de uma análise do estado da arte de soluções e abordagens existentes e uma avaliação das mesmas. Concluindo com um estudo das tecnologias a serem aplicadas no desenvolvimento da solução e na realização da análise de valor da solução que se pretende desenvolver.

### **2.1 GesProj**

Como foi indicado anteriormente, existe um portal desenvolvido como tema de um projeto de PESTI, com o objetivo de facilitar a gestão de projetos, aspetos financeiros, e para cada projeto permitir a gestão das afetações de cada investigador, os seus indicadores de realização e suas atividades. Nesta secção o GesProj será explorado em maior detalhe.

Antes de se abordar a arquitetura do sistema e o portal em si, é realizado um breve estudo do domínio do problema. Começando-se por apresentar os diferentes termos necessários para a compreensão do negócio, através da Tabela 1.

Tabela 1 - Glossário (Couto 2019)

<b>Termo</b>	<b>Significado</b>
Projeto	Projeto de investigação
Orçamento	Orçamento de um projeto
Rúbrica	Termo orçamental que representa tipos de despesa (i.e. viagens, recursos humanos, etc.)
Despesa	Gasto realizado numa certa rubrica.
Investigador	Profissional associado a um projeto.
Afetação	Percentagem de afetação das horas de trabalho de um investigador a um projeto.
Atividade	Ação específica a ser realizada no projeto
Indicador	Indicadores de realização – dizem respeito a publicações, registo de patentes, <i>software</i> produzido, relatórios, etc., produzidos ao longo da execução de um determinado projeto.
Centro de custos	Termo contabilístico que pode representar um departamento, ou área de negócio dentro de uma instituição. Neste caso, o ISEP cria um centro de custos para cada projeto de investigação.
Taxa de financiamento	Percentagem do valor elegível de um projeto que é financiada a fundo perdido pelas entidades financiadoras (i.e.: P2020, H2020, FEDER, etc...)
Custos indiretos	Os custos indiretos compreendem todos os custos elegíveis que não podem ser identificados pelo beneficiário como diretamente imputáveis ao projeto, mas que se encontram relacionados com os custos diretos elegíveis atribuídos ao mesmo.
Pedido de autorização de despesa	É um formulário interno ao ISEP, onde se faz o pedido para adquirir determinado bem, seja equipamento, viagem, inscrição em conferência, etc...

No entanto, os termos e as suas definições não são suficientes para perceber como estas entidades interagem do ponto de vista do negócio, daí apresentar-se o modelo de domínio na Figura 1.

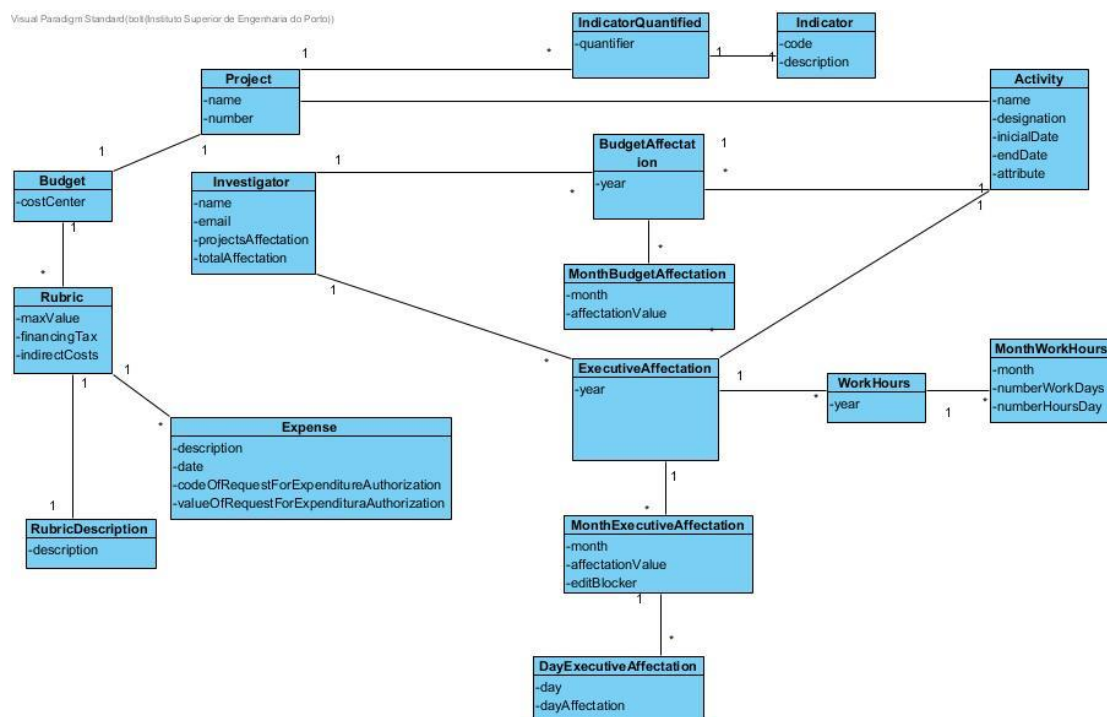


Figura 1 – Modelo de Domínio do GesProj (Couto 2019)

De um ponto de vista do problema tratado por esta dissertação, onde se pretende criar recomendações de investigadores para diferentes candidaturas de projetos, é possível afirmar que a entidade relativa ao projeto (*Project*) e investigador (*Investigator*) serão das mais relevantes. No entanto, estas não são suficientes para sugerir investigadores para um dado projeto visto que não existe relação entre eles, daí ser necessária a entidade atividade (*Activity*).

Com estas três entidades já é possível criar recomendações, visto que as atividades de um projeto podem ser associadas a investigadores, ou seja podem ser sugeridos investigadores para as diferentes atividades.

No entanto o domínio apresentado aqui não só não é o preferível para a realização de recomendações, mas também não é totalmente correto de um ponto de vista do negócio. Primeiro não é considerado o conceito de Candidatura, que representa a fase inicial de um Projeto antes deste ser aprovado e iniciado.

Em relação ao conceito de Atividade, este tem em falta alguns conceitos que permitem melhor a caracterizar. Uma Atividade deve ser caracterizada por uma Área Científica que indica a competência envolvida na sua realização e também tem objetivos, ou *milestones*, que tem de ser atingidos durante a sua realização.

Tal como o conceito de Atividade, o Investigador tem em falta o conceito de Perfil Científico, que indica as Áreas Científicas onde este tem competências. Finalmente, o conceito de Indicador Realizado, pois mesmo que sejam registados os Indicadores que têm de ser realizados (*IndicatorQuantified*), também é necessário registar os indicadores já realizados.

Com o modelo de domínio do portal apresentado, abordam-se agora os diferentes casos de uso que têm como objetivo permitir a um utilizador gerir as diferentes entidades do domínio, através do diagrama de casos de uso da Figura 2.



Figura 2 – Diagrama de Casos de Uso do GesProj (Couto 2019)

Tendo apresentado o diagrama de domínio e casos de uso, aborda-se agora a arquitetura do sistema que foi desenvolvido. Esta segue uma abordagem modular, sendo assim dividido em quatro componentes. Estes componentes e as suas ligações são apresentadas na Figura 3.

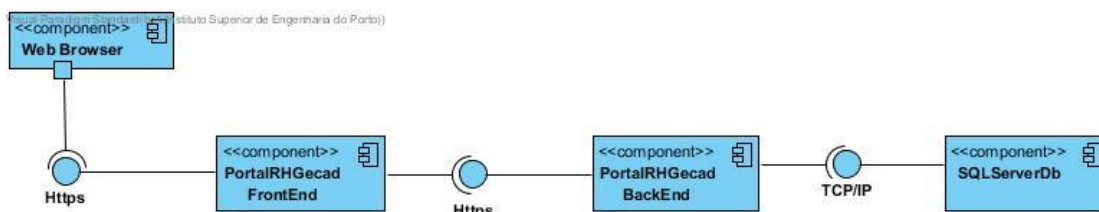


Figura 3 - Diagrama de componentes do GesProj (Couto 2019)

Como indicado pelo nome, o componente Web Browser representa qualquer navegador *web* utilizado para aceder às diferentes funcionalidades do sistema. O componente SQLServerDb representa a camada de persistência do sistema, responsável por guardar toda a informação do mesmo, tendo sido implementada em *SQL Server*.

O componente PortalRHGecadBackEnd, tal como é sugerido pelo nome, é o componente *back-end* do sistema, responsável por ligar a aplicação cliente à base de dados e disponibilizar-lhe diferentes serviços relevantes à realização das diferentes tarefas do negócio. Este componente foi implementado utilizando a *framework ASP.NET Core*.

A aplicação cliente é representada pelo componente PortalRHGecadFrontEnd, esta é responsável por disponibilizar ao utilizador, através do componente Web Browser, acesso às diferentes funcionalidades do sistema. Este componente foi implementado com a utilização da *framework React* (Couto 2019).

Quando um utilizador acede ao portal através do seu browser, ele é recebido pela página apresentada na Figura 4. Não é necessário efetuar o login do utilizador no sistema, visto que esta funcionalidade não se encontra operacional (Couto 2019).

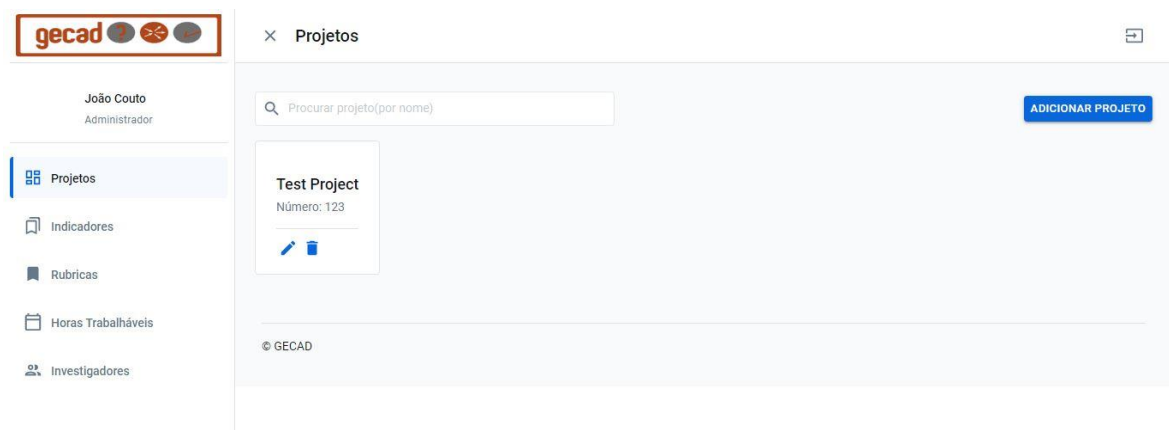


Figura 4 – Página inicial do portal

Como se pode observar no menu à esquerda é possível realizar operações sobre projetos, indicadores, rúbricas, horas trabalháveis e investigadores. No caso da página relativa à gestão de projetos é possível criar projetos ou escolher um projeto já existente e consultar ou alterar a sua informação, como é apresentado na Figura 5.

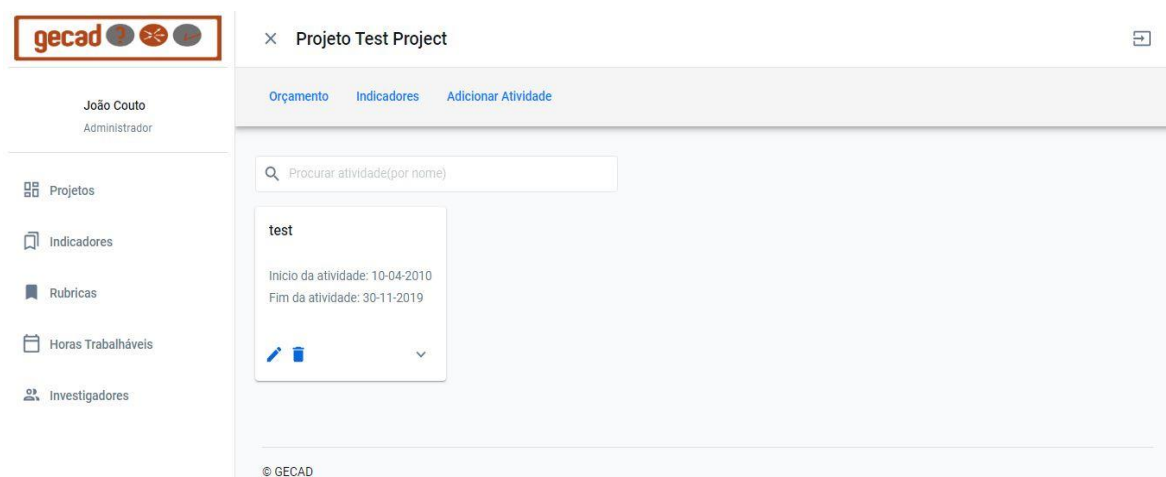


Figura 5 – Página de gestão de um projeto

Escolhendo a opção “Indicadores” no menu, são apresentados os diferentes indicadores existentes no sistema em conjunto com a opção de criar indicadores, como é apresentado na Figura 6. Se um destes indicadores for selecionado é possível eliminá-lo ou editar a sua informação. A página relativa às rúbricas é semelhante ao que é apresentado na Figura 6, onde são listadas as diferentes rúbricas, com a opção de criar ou editar as existentes.

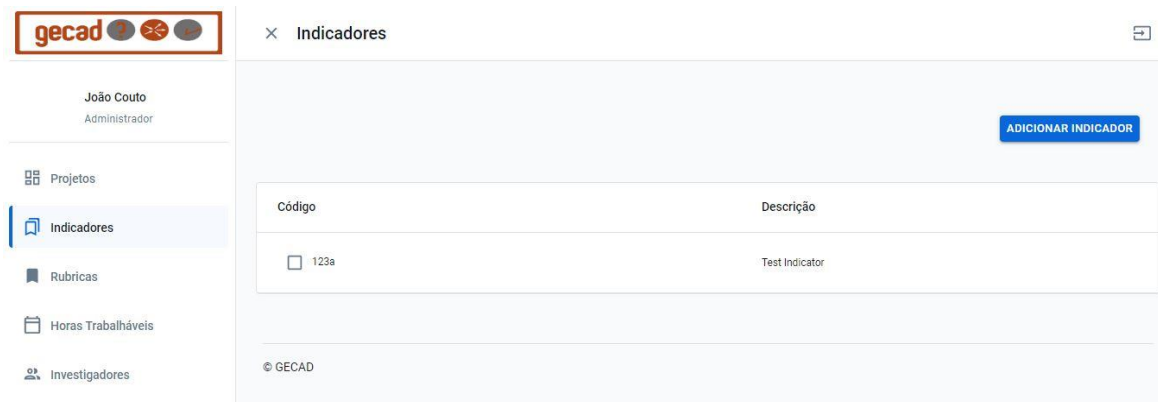


Figura 6 – Página de gestão de indicadores

Escolhendo a opção “Horas Trabalháveis”, são apresentados os diferentes anos registados no sistema, com a opção de registar mais anos, indicando os seus dias trabalhados e jornadas, ou escolher um dos existentes para consultar ou editar a sua informação. Na Figura 7 é apresentada a página em questão.



*Figura 7 – Página de gestão de horas trabalháveis*

Escolhendo um dos anos é apresentada uma tabela com os vários meses, em conjunto com os respetivos dias trabalháveis e jornadas, para consultar a informação ou editar os dias ou jornadas. A página em questão é apresentada na Figura 8.

Mês	Dias Trabalháveis	Jornada
Jan	15	5
Feb	20	7
Mar	20	7
Apr	20	7
May	20	7
Jun	20	7
Jul	20	7

*Figura 8 – Página das horas trabalháveis de um ano*

A última opção permite a gestão de investigadores. Tal como no caso dos indicadores e rúbricas, são listados os diferentes investigadores, com a opção de criar investigadores e editar ou eliminar um dado investigador. Como é apresentado na Figura 9.

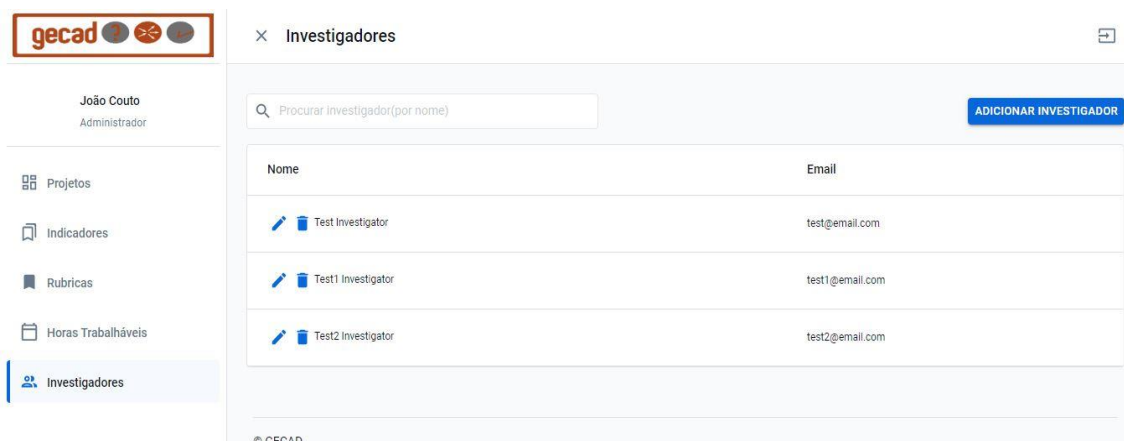


Figura 9 – Página de gestão de investigadores

É possível afetar um investigador a uma atividade de um projeto, no entanto não é possível visualizar ou editar esta informação, o que impossibilita a gestão de folhas de pagamento (Couto 2019).

No âmbito da realização do respetivo relatório de PESTI, no qual a informação desta secção se baseia, as funcionalidades do portal e o seu estado de concretização encontra-se resumidos na Tabela 2.

Tabela 2 - Estado de concretização do portal (Couto 2019)

Objetivos	Estado
Gestão de investigadores	Parcialmente concretizado
Registo e acesso a projetos	Concretizado
Controlo de execução orçamental	Concretizado
Gestão de indicadores	Concretizado
Gestão de afetação dos utilizadores	Parcialmente concretizado
Gestão de folhas de pagamento	Não foi concretizado
Gestão de horas e dias trabalháveis	Concretizado

Neste contexto, esta solução inicial do GesProj será alterada para conseguir interagir com os novos componentes a serem desenvolvidos, o que implicará alterações no seu domínio, arquitetura, e a criação de novas funcionalidades e alteração de funcionalidades já existentes.

## 2.2 Abordagens Existentes

Nesta secção, é realizada uma análise do estado da arte relativo ao problema a ser resolvido. Mais especificamente é realizada uma explicação teórica dos conceitos envolvidos na resolução do problema.

Os sistemas de recomendação são criados com o objetivo de recomendar a um individuo ou grupo, os produtos ou serviços (*items*) mais relevantes. Estas recomendações tendem a ser baseadas na informação do individuo ou grupo a quem as recomendações se destinam, relacionando-as com a informação dos *items* a serem recomendados e interações entres estes (Lu et al. 2015).

Estes sistemas são considerados uma das abordagens mais utilizadas de *Machine Learning* (ML), pois são adotadas por várias entidades como a Amazon, Netflix, entre outras. No entanto ao contrário da maioria dos sistemas de ML, os sistemas de recomendação têm um foco na personalização dos resultados obtidos pelos algoritmos e na importância dada à diversidade dos resultados (Varga 2019).

Tipicamente os sistemas de recomendação recolhem dois tipos de informação (Varga 2019):

- **Informação explícita:** Representa a informação disponibilizada pelos utilizadores, de forma voluntária.
- **Informação implícita:** Representa a informação obtida pelo sistema enquanto este monitoriza as atividades do utilizador.

Em relação à classificação de um sistema de recomendação, nesta dissertação é colocado um foco em três categorias diferentes, dependendo de como as recomendações são geradas (Olszak 2016), podendo assim ser:

- Recomendações baseadas em conteúdo;
- Recomendações colaborativas;
- Abordagens híbridas.

Nas secções 2.2.1 a 2.2.5 são exploradas com maior detalhe três abordagens, bem como as diferentes medidas e métricas aplicadas no seu desenvolvimento. Na secção 2.2.6, é analisada a utilização de sistemas de recomendação na gestão de recursos humanos.

### 2.2.1 Recomendações baseadas em conteúdo

Esta abordagem baseia-se na criação de recomendações com base em preferências antigas do utilizador (Olszak 2016), com o objetivo de determinar os atributos que podem ser utilizados para os distinguir dos restantes (Lu et al. 2015).

Os atributos recolhidos são guardados no perfil do utilizador para depois serem comparados com os atributos dos restantes *items* para assim se poder recomendar aqueles com as maiores semelhanças às preferências do utilizador em questão (Lu et al. 2015).

Na utilização desta abordagem, as recomendações podem ser geradas através de heurísticas, com a utilização de métodos clássicos de extração de informação, ou através de abordagens baseadas em ML, com a criação de modelos capazes de aprender as preferências dos utilizadores através de informação já existente (Lu et al. 2015).

A abordagem em questão é normalmente utilizada quando existe uma grande quantidade de informação disponível para o sistema utilizar. Ou seja, esta abordagem é limitada quando existe pouca informação relativamente ao utilizador ou *item* a ser considerados no momento, sofrendo assim de problemas de *cold-start* (Aggarwal 2016b).

Outro aspeto menos positivo desta abordagem é o facto de não considerar as preferências de outros utilizadores, fazendo com que as recomendações realizadas tornam-se previsíveis e pouco diversas, pois apenas se baseiam no utilizador em questão (Aggarwal 2016b).

### 2.2.2 Recomendações colaborativas

A abordagem de recomendações colaborativas baseia-se na criação de recomendações com base em gostos e preferências semelhantes de outros utilizadores (Olszak 2016). Esta pode ser categorizada segundo duas abordagens distintas (Lu et al. 2015):

- **Abordagem com base nos utilizadores:** Recomendações são geradas com base em preferências semelhantes apresentadas por utilizadores semelhantes.
- **Abordagem com base em *items*:** Recomendações são geradas com base em *items* semelhantes aos preferidos pelo utilizador no passado.

A maioria dos sistemas de recomendação colaborativa baseiam-se em medições numéricas da preferência de um utilizador por um *item*, ou seja, o seu *rating*. Como por exemplo o *rating* que é possível apresentar por produtos em sistemas como o da Amazon, com base numa escala de 5 estrelas (Kluver, Ekstrand, e Konstan 2018).

No entanto, devido ao facto da maioria dos utilizadores não apresentar *ratings*, ou preferências, por todos os *items* do sistema, existem casos onde a informação de certos *items* é escassa (Aggarwal 2016a). Mesmo assim, é possível representar os valores dos *ratings* não especificados, visto que as medições já determinadas estão correlacionadas através dos vários utilizadores e *items* (Aggarwal 2016a).

Ou seja, quando se consideram os *ratings* de utilizadores com preferências semelhantes, é possível inferir o *rating* para um *item* não considerado por um dado utilizador, através dos gostos daqueles que tem preferências semelhantes às suas (Aggarwal 2016a).

Quanto à implementação desta técnica, tal como na secção anterior, são utilizadas abordagens com base em memória ou modelos.

Abordagens baseadas em memória também são designadas como algoritmos de “*neighborhood based collaborative filtering*”, onde combinações entre utilizadores e *items* são determinadas com base na sua proximidade, ou *neighborhood*. Se esta se focar nos utilizadores, as recomendações feitas a um dado utilizador são criadas com base nas preferências de utilizadores com interesses semelhantes (Aggarwal 2016a).

No caso de uma abordagem baseada em memória com foco em *items*, começa-se com a criação de previsões face ao interesse de um dado utilizador por um *item*. Tal processo é realizado através da recolha de um conjunto de *items* semelhantes ao *item* a ser considerado, o *rating* apresentado pelo utilizador pelos elementos do conjunto são depois considerados para determinar o interesse do mesmo no *item* em questão (Aggarwal 2016a).

A abordagem com base em modelos, “ML e *data mining* são utilizados no contexto de métodos preditivos. Se o modelo for parametrizável, os parâmetros são aprendidos no contexto de *frameworks* de otimização”. Dos métodos disponíveis, existem árvores de decisão, modelos baseados em regras, entre outros, onde alguns conseguem obter bons resultados mesmo com informação escassa nas matrizes de *ratings* (Aggarwal 2016a).

### **2.2.3 Abordagens híbridas**

Tal como o nome desta técnica indica, este método baseia-se numa aplicação conjunta dos métodos apresentados anteriormente, nas secções de Recomendações baseadas em conteúdo e Recomendações colaborativas (Olszak 2016).

Sistemas de recomendação híbridos podem ser classificados segundo várias categorias, como é apresentado de seguida na Figura 10:

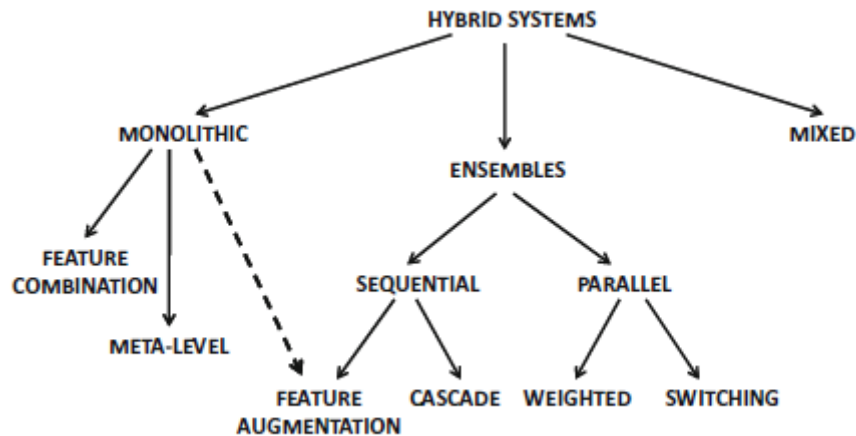


Figura 10 - Categorias de sistemas de híbridos de recomendação (Aggarwal 2016c)

Primeiro aborda-se as categorias colocadas no primeiro nível da figura apresentada:

- **Monolithic:** Sistemas de recomendação existentes são alterados, ou criados de raiz, através da combinação de características de diferentes tipos de dados.
- **Ensemble:** São geralmente estruturados através da combinação paralela ou sequencial de diferentes sistemas de recomendação.
- **Mixed:** São apresentadas as recomendações geradas por diferentes sistemas ao mesmo tempo.

A um nível mais aprofundado, estes sistemas podem ainda ser classificados com base num maior número de categorias (Aggarwal 2016c), sendo estas:

- **Weighted:** As pontuações dos vários componentes do sistema são determinadas e combinadas utilizando uma combinação linear.
- **Switching:** O sistema alterna entre diferentes técnicas de recomendação com base nas necessidades atuais.
- **Cascade:** As recomendações obtidas por uma técnica de recomendação são aperfeiçoadas por outra.
- **Feature augmentation:** “O output de uma técnica é utilizado para criar o input de uma outra técnica”(Aggarwal 2016c).

- **Feature combination:** “Características de diferentes fontes de dados são combinadas e utilizadas no contexto de um único sistema de recomendação” (Aggarwal 2016c).
- **Meta-level:** O modelo utilizado por uma técnica de recomendação é utilizado como *input* de outra.
- **Mixed:** São apresentadas as recomendações geradas por diferentes técnicas ao mesmo tempo.

#### 2.2.4 Medidas de Semelhança

Medidas de Semelhança são um conjunto de métodos utilizados na determinação das semelhanças entre utilizadores e *items*. A escolha da medida de semelhança a ser utilizada é muito importante, pois influencia a capacidade do sistema de recomendação apresentar resultados precisos (Adomavicius, Manouselis, e Kwon 2011; Agarwal e Chauhan 2017). Nesta secção serão abordadas algumas das Medidas de Semelhança existentes.

A **Correlação de Pearson**, é a abordagem mais utilizada e permite calcular a correlação linear entre dois vetores. Os resultados obtidos por este cálculo encontram-se no intervalo [-1;1]. Onde 1 representa uma correlação positiva, -1 uma correlação negativa e 0 a falta de correlação (Agarwal e Chauhan 2017). Esta correlação pode ser enunciada pela seguinte equação:

$$PCC\_Sim(u, v) = \frac{\sum_{i \in I} (r_{u,i} - \bar{r}_u, I)(r_{v,i} - \bar{r}_v, I)}{\sqrt{\sum_{i \in I} (r_{u,i} - \bar{r}_u, I)^2 + \sum_{i \in I} (r_{v,i} - \bar{r}_v, I)^2}} \quad (1)$$

Onde  $\bar{r}_u, I$  e  $\bar{r}_v, I$  representam a “média dos *ratings* dos utilizadores  $u$  e  $v$ , pelos produtos representados por  $I$ ” (Agarwal e Chauhan 2017).

A **Semelhança do Cosseno**, também é uma das abordagens mais utilizadas e permite determinar o ângulo cosseno entre dois vetores. No entanto esta apresenta a falha de considerar preferências nulas como negativas (Agarwal e Chauhan 2017). Esta semelhança pode ser enunciada pela seguinte equação:

$$Cos\_Sim(u, v) = \frac{\vec{R}_u \cdot \vec{R}_v}{\|\vec{R}_u\| \|\vec{R}_v\|} \quad (2)$$

Onde “.” representa o “produto entre os vetores  $\vec{R}_u$  e  $\vec{R}_v$  representativos dos *ratings* apresentados pelos utilizadores  $u$  e  $v$ ” (Agarwal e Chauhan 2017).

A **Semelhança de Jaccard**, considera as preferências comuns entre utilizadores para calcular a semelhança entre estes. A semelhança entre dois utilizadores é tanto maior quanto maior o número de escolhas comuns entre si. No entanto esta “produz um número limitado de valores que dificulta a diferenciação de utilizadores” (Agarwal e Chauhan 2017). Esta semelhança pode ser enunciada pela seguinte equação:

$$Jaccard\_sim(u, v) = \frac{|I_u \cap I_v|}{|I_u \cup I_v|} \quad (3)$$

Onde  $I_u$  e  $I_v$  representam os conjuntos dos *items* pelos quais o utilizador  $u$  e  $v$  apresentaram preferência, respetivamente.

### 2.2.5 Métricas de Avaliação

No desenvolvimento de sistemas de recomendação são necessárias métricas de avaliação e qualidade para determinar a validade dos algoritmos, técnicas e métodos escolhidos para o desenvolvimento do sistema (Sarwar et al. 2001; Singh e Solanki 2019). Estas métricas podem ser classificadas segundo duas categorias.

Existe a categoria de Métricas de Precisão Estatística que avaliam a precisão do sistema comparando as recomendações geradas com os *ratings* apresentados pelo utilizador (Sarwar et al. 2001).

O *Mean Absolute Error* (MAE), ou Erro Médio Absoluto, é uma métrica muito utilizada que mede a “variação das recomendações dos *ratings* apresentados pelos utilizadores”. Quanto mais baixo o MAE melhor é o sistema (Sarwar et al. 2001). Esta métrica pode ser enunciada pela equação:

$$MAE = \sum \frac{|p_{i,j} - r_{i,j}|}{n} \quad (4)$$

Onde  $p_{i,j}$  é a recomendação do item  $j$  para o utilizador  $i$ , enquanto que  $r_{i,j}$  é o *rating* apresentado pelo utilizador  $i$ ,  $n$  representa o total de pares recomendação-*rating* (Singh e Solanki 2019).

Outra métrica também utilizada é a *Root Mean Squared Error* (RMSE), ou Raiz Quadrada do Erro Quadrático Médio, esta representa o desvio entre os valores dos *ratings* e os valores das recomendações (Singh e Solanki 2019). Esta métrica pode ser enunciada pela seguinte equação:

$$RMSE = \sqrt{\frac{\sum (p_{i,j} - r_{i,j})^2}{n}} \quad (5)$$

Onde  $p_{i,j}$  é a recomendação do *item*  $j$  para o utilizador  $i$ , enquanto que  $r_{i,j}$  é o *rating* apresentado pelo utilizador  $i$ ,  $n$  representa o total de pares recomendação-*rating* (Singh e Solanki 2019).

Também existe a categoria de Métricas de Precisão do Apoio à Decisão que avaliam a capacidade do sistema de recomendação ajudar o utilizador a escolher os melhores *items*. Esta métricas assumem que o processo de recomendação é uma operação binária com *items* “bons” e “maus” (Sarwar et al. 2001).

A Precisão, é uma das métricas mais utilizadas nesta categoria, responsável por determinar do total de *items* recomendados quantos são de facto relevantes (Google 2018; Singh e Solanki 2019). Esta métrica pode ser enunciada pela seguinte equação:

$$Precisão = \frac{N^{\circ} \text{ de } items \text{ relevantes recomendados}}{N^{\circ} \text{ de } items \text{ recomendados}} \quad (6)$$

O *Recall*, também é uma métrica muito utilizada, responsável por determinar do total de *items* considerados relevantes quantos foram recomendados (Google 2018; Singh e Solanki 2019). Esta métrica pode ser enunciada pela seguinte equação:

$$Recall = \frac{N^{\circ} \text{ de } items \text{ relevantes recomendados}}{N^{\circ} \text{ total de } items \text{ relevantes}} \quad (7)$$

A *F-Measure*, é a combinação da Precisão e do *Recall* (Singh e Solanki 2019) como é apresentado na seguinte equação:

$$F - Measure = \frac{2 \times Precisão \times Recall}{(Precisão + Recall)} \quad (8)$$

### 2.2.6 Recomendação de Recursos Humanos para Equipas

Como foi mencionado no início do capítulo um dos aspetos do problema em questão assenta na necessidade de sugerir recursos humanos para um dado projeto, ou seja, este aspeto encontra-se relacionado com as áreas de recomendação de equipas e a gestão de recursos humanos.

Um sistema de recomendação de equipas pode ser uma grande ajuda para uma empresa, pois ajuda gestores a criarem equipas para uma dada tarefa, no entanto o processo de criação da equipa em si é complexo (Datta, Braghin, e Yong 2013; Malinowski, Weitzel, e Keim 2008).

Tal complexidade vem do facto de para lá de considerar a capacidade de uma pessoa realizar uma dada tarefa, também é necessário considerar aspetos humanos. Aspetos como as características individuais dos vários elementos da equipa e as interações que ocorrem entre estes, que depois influenciam a capacidade da equipa como um todo (Malinowski, Weitzel, e Keim 2008).

Ou seja, o processo de recomendação de elementos para equipas é complexo pois considera diferentes aspetos relativos aos indivíduos e às tarefas, existindo assim um foco nas características *person-job* (P-J) e *person-team* (P-T) *fit* (Bell, Brown, e Weiss 2018; Malinowski, Weitzel, e Keim 2008).

- **P-J fit:** Representa a relação entre o indivíduo e a tarefa, em termos das características da tarefa face às capacidades do indivíduo e as preferências do indivíduo (Malinowski, Weitzel, e Keim 2008).
- **P-T fit:** Representa a relação entre o indivíduo e a equipa, pois é necessário considerar como este interage com outras pessoas (Malinowski, Weitzel, e Keim 2008).

Tanto abordagens baseadas em P-J e P-T *fit* são consideradas “importantes para prever a *performance* de uma equipa, e podem ser incorporadas em decisões de seleção ou colocação, de forma formal ou informal” (Bell, Brown, e Weiss 2018).

Das abordagens mencionadas na literatura uma solução pode ter em consideração todos estes aspetos relacionados com a criação de um equipa, ou apenas alguns como as características P-J *fit* (Malinowski, Weitzel, e Keim 2008). No contexto desta dissertação é este tipo de solução que se considera como interessante, pois apenas se pretende sugerir investigadores para uma candidatura de projeto face ao seu perfil científico.

O mesmo se verifica num contexto profissional, pois um grande número de organizações tende a utilizar abordagens baseadas em P-J *fit*. Devido ao facto de estas colocarem um maior foco na escolha de indivíduos, ignorando as complexidades relacionadas com as interdependências de uma equipa (Bell, Brown, e Weiss 2018).

No entanto é necessário mencionar que estas soluções, com um foco no P-J *fit*, são consideradas como soluções “incompletas”, tendo assim diferentes desvantagens, pois ignoram os aspetos de P-T *fit* (Bell, Brown, e Weiss 2018; Malinowski, Weitzel, e Keim 2008).

Independentemente de como são implementados estes sistemas têm como objetivo sugerir as “melhores” equipas, no entanto o conceito de “melhor” é considerado subjetivo, visto que varia conforme as métricas utilizadas para avaliar as soluções (Datta, Braghin, e Yong 2013).

## 2.3 Soluções Existentes

Nesta secção apresenta-se soluções existentes de implementações e aplicações de sistemas de recomendação.

### 2.3.1 ReSySTER

O ReSySTER é um sistema de recomendação com base numa abordagem híbrida e na aplicação de lógica *fuzzy*, teoria de *rough sets* e tecnologias semânticas. Este sistema pretende disponibilizar a gestores de projetos uma ferramenta capaz de ajudar no processo de criação de equipa em ambientes *Scrum*. Para tal existem dois tipos de utilizador no sistema, o gestor de projetos que pede as recomendações e o perito que tem a responsabilidade de manter o sistema calibrado (Colomo Palacios et al. 2012).

Para associar elementos a equipas, o ReSySTER baseia-se nas competências envolvidas na realização de cada *Work Package* (WP) do projeto e nas competências de cada trabalhador disponível. Quando um gestor de projetos pretende receber recomendações de equipas para cada WP, ele pode definir o número de indivíduos de cada equipa e o nível de competência (de 0 a 100) envolvido na realização da WP em questão (Colomo Palacios et al. 2012).

Com base nos parâmetros apresentados pelo gestor, o sistema aplica transformações *fuzzy*, categorização *rough set* e *matching*, para determinar quais dos trabalhadores considerados são os melhores para fazerem parte da equipa que no fim do processo é sugerida ao gestor (Colomo Palacios et al. 2012).

### 2.3.2 Social Web Application for Team Recommendation (SWAT)

SWAT é uma *framework* para a recomendação de equipas multidisciplinares, que se diferencia de muitas outras soluções por ser uma solução genérica não se focando em nenhum problema

em específico, podendo assim ser utilizada em diferentes contextos (Datta, Braghin, e Yong 2013).

Para utilizar SWAT é necessário obter e tratar informação relativa aos indivíduos que vão ser considerados nas recomendações, informação de áreas de experiência e informação das relações existentes entre os diferentes indivíduos e entre as áreas de competência. Esta informação forma o corpus que posteriormente vai ser utilizado no processo de recomendação (Datta, Braghin, e Yong 2013).

Quanto ao processo de recomendação em si, este é baseado em objetivos, que neste contexto são identificados por um dado conjunto de áreas de competência. Como pode haver diferentes combinações de indivíduos que possuem as competências necessárias, a SWAT disponibiliza quatro métricas para melhor definir as características da equipa que o utilizado necessita. As métricas são as seguintes (Datta, Braghin, e Yong 2013):

- **Pontuação de Competência:** Representa o grau de competência dos indivíduos face ao nível de competência das áreas de competências envolvidas na realização do objetivo;
- **Pontuação de Coesão Social:** A métrica representa as características sociais da equipa, tomando em consideração as relações sociais existentes entre os indivíduos;
- **Repetição de Elementos da Equipa:** A métrica é utilizada para determinar a probabilidade de certos indivíduos trabalharem bem em conjunto;
- **Repetição de Conceitos da Equipa:** A métrica é utilizada para determinar, com base em equipas antigas, a probabilidade da equipa conseguir lidar com as competências do objetivo atual.

### 2.3.3 Opportunity Team Builder (OTB)

OTB é uma solução aplicada na recomendação de equipas de venda, para que estas depois possam ser utilizadas para vender um produto ou conjunto de produtos. Para atingir este objetivo a solução determina quais os papéis necessários para a equipa realizar uma venda, recomenda indivíduos com base na sua capacidade de realizar estes papéis e depois prevê o impacto de adicionar membros à equipa na realização da venda (Alkan, Daly, e Vejsbjerg 2018).

A solução determina quais os vendedores a sugerir papel a papel, considerando para cada papel vendedores com base na sua experiência técnica, conhecimento do sector e ligações pessoais com os restantes elementos de equipa. O utilizador tem a capacidade de especificar papéis a serem considerados na geração de recomendações, alterar as recomendações apresentadas

podendo remover/adicionar elementos, ou realizar outras alterações que levem a alterações da *performance* da equipa como um todo (Alkan, Daly, e Vejsbjerg 2018).

Mesmo que de momento a OTB esteja apenas a ser utilizada na recomendação de equipas de venda, no futuro pode vir a ser alterada para ser aplicada noutros contextos (Alkan, Daly, e Vejsbjerg 2018).

## **2.4 Avaliar Soluções e Abordagens Existentes**

Nesta secção é realizada uma avaliação das soluções e abordagens existentes de modo a determinar quais são as mais indicadas a serem utilizadas nesta dissertação. Para realizar a avaliação das abordagens existentes de sistemas de recomendação foi escolhido o método *Analytic Hierarchy Process* (AHP). Quanto à avaliação das soluções existentes decidiu-se utilizar uma tabela comparativa, entre as três soluções apresentadas e as características desejadas na solução.

### **2.4.1 AHP**

O método AHP, foi definido em 1988 por Thomas L Saaty, com o objetivo de assistir processos de decisão complexos, que normalmente seriam realizados de forma reativa e não planeada. Com a aplicação de AHP, é possível considerar diferentes fatores para a realização de uma dada decisão através da aplicação de lógica matemática (Saaty 1988).

Para a sua aplicação neste contexto será utilizada a escala de classificação do nível de importância de cada critério definido, presente na Figura 11.

Nível de importância	Definição	Explicação
1	Igual importância	As duas atividades contribuem igualmente para o objetivo
3	Fraca importância	A experiência e o julgamento favorecem levemente uma atividade em relação à outra
5	Forte importância	A experiência e o julgamento favorecem fortemente uma atividade em relação à outra
7	Muito forte importância	Uma atividade é muito fortemente favorecida em relação a outra
9	Importância absoluta	A evidência favorece uma atividade em relação a outra com o mais alto grau de certeza
2,4,6,8	Valores intermediários	Quando se procura uma condição de compromisso entre duas definições

Figura 11 - Níveis de importância de comparações (Nicola 2019; Saaty 1990)

Para a realização do cálculo da razão de consistência (RC) é utilizado o IR sendo este um índice aleatório calculado para matrizes quadradas de ordem  $n$ . Os valores IR a serem considerados neste contexto são apresentados na Figura 12.

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
0.00	0.00	0.58	0.90	1.12	1.24	1.32	1.41	1.45	1.49	1.51	1.48	1.56	1.57	1.59

Figura 12 – Valores IR (Nicola 2019; Saaty 1990)

#### 2.4.2 Avaliação de Abordagens

Para esta avaliação utilizou-se o método AHP para auxiliar a decisão de qual a abordagem mais apropriada a ser aplicada na solução proposta com base naquelas estudadas anteriormente:

- Recomendações baseadas em conteúdo;
- Recomendações colaborativas;
- Abordagens híbridas.

Para a aplicação do método AHP, é necessário definir critérios pelos quais comparar as três possibilidades. Para o efeito foram definidos 3 critérios:

- *Cold-Start*: Representa a capacidade da abordagem ser capaz de lidar com a falta de informação relativamente a um utilizador ou *item*;
- Variabilidade: Representa a capacidade da solução ser capaz de produzir uma grande diversidade de recomendações;

- *Performance*: Representa o tempo que a abordagem irá demorar a criar recomendações com base num dado *input*.

Tendo definido os critérios começa-se pela construção da árvore hierárquica de decisão para a escolha de um sistema de recomendação. Esta árvore é apresentada na Figura 13.

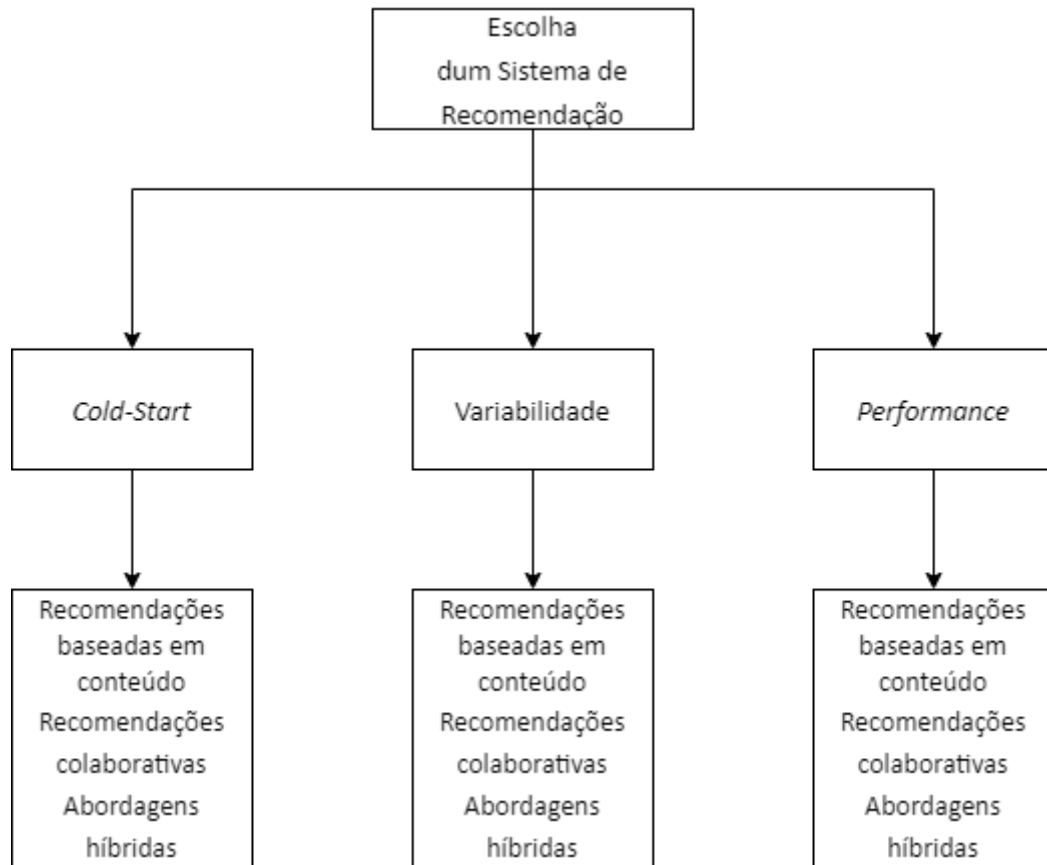


Figura 13 - Árvore Hierárquica

Com a criação da árvore hierárquica, segue-se para a criação de uma tabela comparativa entre os diferentes critérios. Assume-se assim que a *Performance* é o critério com o menor valor para a realização da decisão e a *Variabilidade* é ligeiramente mais importante que o critério de *Cold-Start*. Com estas considerações é criada a Tabela 3, onde também se encontra presente a soma das suas colunas.

Tabela 3 – Tabela Comparativa de Critérios

	<b>Cold-Start</b>	<b>Variabilidade</b>	<b>Performance</b>
<i>Cold-Start</i>	1	1/2	3
Variabilidade	2	1	4
<i>Performance</i>	1/3	1/4	1
<i>Soma das Colunas</i>	10/3	7/4	8

Após a criação da tabela esta tem de ser normalizada, para depois ser possível o cálculo da prioridade relativa de cada critério. A normalização é determinada através da divisão de cada elemento da tabela pela soma resultante da sua respetiva coluna. Enquanto que a prioridade relativa é obtida através da média aritmética de cada linha da tabela. Na Tabela 4, são apresentados os valores normalizados em conjunto com a respetiva prioridade relativa.

Tabela 4 - Tabela Comparativa de Critérios Normalizada

	<b>Cold-Start</b>	<b>Variabilidade</b>	<b>Performance</b>	<b>Prioridade Relativa</b>
<i>Cold-Start</i>	3/10	2/7	3/8	0.32024
Variabilidade	3/5	4/7	1/2	0.55714
<i>Performance</i>	1/10	1/7	1/8	0.12262

A coluna relativa às prioridades relativas representa o vetor de prioridades, ou vetor próprio, que indica o peso de cada critério e confirma que de facto a Variabilidade é o critério mais significativo, seguido do *Cold-Start* e finalmente a *Performance*.

Seguidamente calcula-se o valor da RC, para determinar a consistência dos valores das prioridades relativas. No entanto é necessário determinar o valor de  $\lambda_{\max}$  através da fórmula  $Ax = \lambda_{\max}x$  que determina o maior valor próprio da matriz A. Onde x é o vetor de prioridades relativas, e A é a Tabela 3 sobre a forma de matriz.

$$\begin{bmatrix} 1 & \frac{1}{2} & 3 \\ 2 & 1 & 4 \\ \frac{1}{3} & \frac{1}{4} & 1 \end{bmatrix} \begin{bmatrix} 0.32024 \\ 0.55714 \\ 0.12262 \end{bmatrix} \cong \lambda_{max} \begin{bmatrix} 0.32024 \\ 0.55714 \\ 0.12262 \end{bmatrix}$$

$$\begin{bmatrix} 0.96667 \\ 1.6881 \\ 0.36865 \end{bmatrix} \cong \lambda_{max} \begin{bmatrix} 0.32024 \\ 0.55714 \\ 0.12262 \end{bmatrix}$$

$$\lambda_{max} = average\{0.96667/0.32024, 1.6881/0.55714, 0.36865/0.12262\} = 3.01832$$

Com o valor de  $\lambda_{max}$  é possível calcular o valor do Índice de Consistência (IC), através da fórmula

$$IC = \frac{(\lambda_{max} - n)}{n - 1}, \text{ onde } n \text{ é o número de linhas da matriz. Logo } IC = \frac{3.01832 - 3}{3 - 1} = 0.00916.$$

No caso da determinação do IR, este é relativamente fácil de determinar, visto que a matriz de critérios é de ordem 3, logo através da tabela na Figura 12 é possível concluir que o valor de IR é 0.58. Sendo assim já é possível calcular o valor de RC.

$$RC = \frac{IC}{IR} = \frac{0.00916}{0.58} \cong 0.02$$

Sabendo que RC tem o valor de 0.02 e este é menor que 0.1, é possível concluir que os valores das prioridades relativas dos critérios são consistentes.

Com a realização da matriz comparativa dos critérios, e sua validação segue-se a criação da matriz comparativa para cada critério com base nas alternativas consideradas.

Tabela 5 - Tabela Comparativa de Cold-Start

<b>Cold-Start</b>	<b>Recomendações baseadas em conteúdo</b>	<b>Recomendações colaborativas</b>	<b>Abordagens híbridas</b>
Recomendações baseadas em conteúdo	1	1/3	1/3
Recomendações colaborativas	3	1	1
Abordagens híbridas	3	1	1
Soma das Colunas	7	7/3	7/3

Tabela 6 - Tabela Comparativa de Cold-Start Normalizada

<b>Cold-Start</b>	<b>Recomendações baseadas em conteúdo</b>	<b>Recomendações colaborativas</b>	<b>Abordagens híbridas</b>	<b>Prioridade Relativa</b>
Recomendações baseadas em conteúdo	1/7	1/10	2/17	0.12017
Recomendações colaborativas	3/7	3/7	3/7	0.42857
Abordagens híbridas	3/7	3/7	3/7	0.42857

Tabela 7 - Tabela Comparativa de Variabilidade

<b>Variabilidade</b>	<b>Recomendações baseadas em conteúdo</b>	<b>Recomendações colaborativas</b>	<b>Abordagens híbridas</b>
Recomendações baseadas em conteúdo	1	1/3	1/4
Recomendações colaborativas	3	1	1/3
Abordagens híbridas	4	3	1
Soma das Colunas	8	13/3	19/12

Tabela 8 - Tabela Comparativa de Variabilidade Normalizada

<b>Variabilidade</b>	<b>Recomendações baseadas em conteúdo</b>	<b>Recomendações colaborativas</b>	<b>Abordagens híbridas</b>	<b>Prioridade Relativa</b>
Recomendações baseadas em conteúdo	1/8	1/13	3/19	0.11994
Recomendações colaborativas	3/8	3/13	4/19	0.2721
Abordagens híbridas	4/8	9/13	12/19	0.60796

Tabela 9 - Tabela Comparativa de Performance

<b>Performance</b>	<b>Recomendações baseadas em conteúdo</b>	<b>Recomendações colaborativas</b>	<b>Abordagens híbridas</b>
Recomendações baseadas em conteúdo	1	2	4
Recomendações colaborativas	1/2	1	2
Abordagens híbridas	1/4	1/2	1
Soma das Colunas	7/4	7/2	7

Tabela 10 - Tabela Comparativa de Performance Normalizada

<b>Performance</b>	<b>Recomendações baseadas em conteúdo</b>	<b>Recomendações colaborativas</b>	<b>Abordagens híbridas</b>	<b>Prioridade Relativa</b>
Recomendações baseadas em conteúdo	4/7	4/7	4/7	0.57143
Recomendações colaborativas	2/7	2/7	2/7	0.28571
Abordagens híbridas	1/7	1/7	1/7	0.14286

Após ter realizado o cálculo dos diferentes vetores prioridade, segue-se a determinação da melhor abordagem através do cálculo da prioridade composta para das alternativas e escolha da prioridade com o maior valor.

$$\begin{bmatrix} 0.12017 & 0.11994 & 0.57143 \\ 0.42857 & 0.2721 & 0.28571 \\ 0.42857 & 0.60796 & 0.14286 \end{bmatrix} \times \begin{bmatrix} 0.32024 \\ 0.55714 \\ 0.12262 \end{bmatrix} = \begin{bmatrix} 0.17538 \\ 0.32388 \\ 0.49348 \end{bmatrix}$$

Considerando os resultados obtidos, é fácil concluir que a melhor opção é “Abordagens híbridas”.

### 2.4.3 Avaliação de Soluções

Seguidamente é apresentada a Tabela 11, onde as três soluções estudadas na secção de Soluções Existentes são comparadas com base nas características que o GECAD pretende ter na solução a ser desenvolvida.

Tabela 11 – Tabela de Avaliação de Soluções

Características desejadas	ReSySTER	SWAT	OTB
Recomendações baseadas nas capacidades dos indivíduos	✓	✓	✓
Recomendações limitadas por afetações atuais dos indivíduos	✓	✓	✓
Recomendações baseadas nas competências envolvidas na realização do objetivo/projeto	✓	✓	✓
Capacidade do utilizador especificar as competências a serem consideradas na geração de recomendações	✗	✓	✓
Capacidade do utilizador especificar o limite de afetação de cada indivíduo	✗	✗	✗
Capacidade do utilizador especificar o número de indivíduos a ser recomendado	✓	✗	✗

Observando os resultados obtidos da comparação, não existe propriamente uma solução que seja melhor que outra, apenas com base nas características desejadas. Mesmo que estas soluções não sejam utilizadas nesta dissertação, a única que poderia vir a ser utilizada seria a SWAT devido à sua natureza não especializada.

Sendo assim pode-se concluir que mesmo que, segundo a Tabela 11, não exista uma solução que se distinga claramente das outras, a SWAT seria a mais apropriada para o contexto desta dissertação.

## 2.5 Tecnologias

Com base nos requisitos apresentados pelo cliente, foi estabelecido que quanto à implementação das novas funcionalidades ou módulos seriam utilizadas as mesmas tecnologias mencionadas na secção GesProj, ou seja, *ASP.NET Core* e *React*. Mesmo assim é realizada uma análise destas tecnologias com o objetivo de justificar a sua escolha e aplicação neste contexto, para lá da vontade do cliente.

A *framework ASP.NET Core* é uma ferramenta *open-source*, multiplataforma e altamente documentada desenvolvida pela Microsoft e a sua comunidade. Esta é utilizada por vários

desenvolvedores no desenvolvimento de aplicações *web* devido à sua grande flexibilidade na escolha a nível de infraestruturas, base de dados e interação com *frameworks* utilizadas em aplicações cliente. Contem tem suporte para injeção de dependências, que melhora a testabilidade e qualidade do desenvolvimento (Roth, Anderson, e Shaun 2019; Zealous System 2018).

Em conjunto com o desenvolvimento *web*, a *framework* também tem capacidades de desenvolvimento de sistemas de recomendação através da utilização dos *packages* Microsoft.ML e Microsoft.ML.Recommender que disponibilizam as ferramentas para o efeito, incluindo o tratamento de dados e treino do modelo (Warren et al. 2019).

*React* é uma biblioteca JavaScript, *open-source* desenvolvida pelo Facebook, com o foco no desenvolvimento de aplicações cliente para desenvolvimento *web* (TechMagic 2018). Esta é declarativa disponibilizando métodos simples no desenvolvimento de *interfaces* através da utilização de vistas que o *React* consegue atualizar e desenhar com a alteração de dados ao longo do desenvolvimento. Adota uma abordagem baseada em componentes, onde cada um tem a sua responsabilidade e pode interagir com outros para criar *interfaces* mais complexas. Finalmente esta permite desenvolver novas funcionalidades sem alterar código existente (Facebook 2019).

Enquanto que o *React* é uma ferramenta apropriada para ser utilizada nesta solução o mesmo não se pode dizer da *framework ASP.NET Core*, mesmo com a utilização de *packages* relacionados com ML e a geração de recomendações. Uma alternativa apropriada seria *Python*, pois é uma linguagem simples onde existe um grande número de bibliotecas e *frameworks* relacionadas com ML, como por exemplo *TensorFlow* e *Scikit-learn* (Beklemysheva 2018). Estas bibliotecas e *frameworks* podem ser aplicadas no desenvolvimento de diferentes soluções, como por exemplo sistemas de recomendação.

## 2.6 Análise de Valor

Nesta secção é realizada a análise de valor sobre a solução a ser desenvolvida para o problema em questão. É realizada uma introdução ao modelo utilizado, seguida da enunciação da proposta de valor e o valor para o cliente.

### 2.6.1 Modelo de Desenvolvimento de Novos Conceitos

O modelo de Desenvolvimento de Novos Conceitos, ou *New Concept Development* (NCD), consiste numa linguagem comum e meio de compreensão das atividades relacionadas com o *Fuzzy Front End* (FFE) da inovação (P. Koen et al. 2001). Este é composto por três secções que são apresentados na Figura 14.

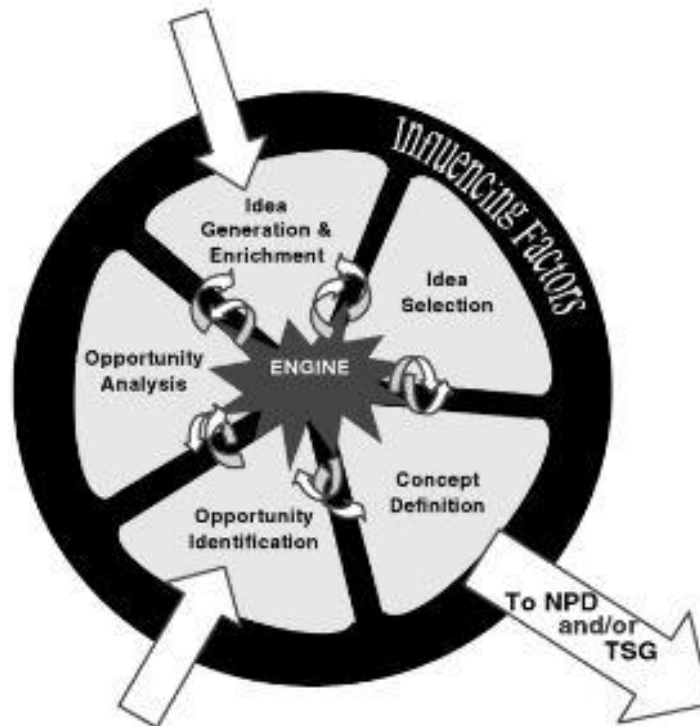


Figura 14 – Modelo NCD (P. A. Koen et al. 2002)

A secção central do modelo é referida como o Motor (*Engine*) e é responsável por gerir os restantes cinco componentes do modelo. O Motor representa a “liderança, cultura e estratégia de negócio” da organização (P. A. Koen et al. 2002).

A camada exterior representa os Fatores influenciadores (*Influencing Factors*). Estes são um conjunto de diferentes fatores, dificilmente controláveis pela organização, responsáveis por afetar o processo de inovação (P. A. Koen et al. 2002).

Entre os Fatores influenciadores e o Motor encontram-se os cinco componentes controláveis do FFE da inovação (P. A. Koen et al. 2002):

- Identificação da Oportunidade (*Opportunity Identification*);
- Análise da Oportunidade (*Opportunity Analysis*);

- Criação e Enriquecimento da Ideia (*Idea Generation & Enrichment*);
- Seleção da Ideia (*Idea Selection*);
- Definição do Conceito (*Concept Definition*).

No contexto do projeto referido neste documento, a etapa de Identificação da Oportunidade ocorreu com a identificação de características que podem ser consideradas como estando em falta no portal do GECAD. Pois enquanto que o portal permite gerir diferentes aspetos relacionados com a gestão de projetos e os seus recursos, não facilita aspetos relativos relacionados com a criação de equipas e a consulta da informação de projetos. Logo num contexto onde em 2018-2019 foram desenvolvidos 33 projetos, cujas equipas tiveram de ser determinadas manualmente e cuja informação é gerida através de folhas de Excel, o portal não facilita ou automatiza a situação.

Tendo identificado estas características, e realizado uma análise de como problemas semelhantes são resolvidos, verificou-se que existem vários artigos e soluções que abordam os conceitos de criação de equipas e consulta de informação agregada. Este processo corresponde à Análise da Oportunidade.

Com a conclusão da Análise da Oportunidade, foi realizada a Criação e Enriquecimento da Ideia onde foi realizado um estudo relativo a soluções aplicadas ao problema em questão. Também foram realizadas várias reuniões com os representantes do GECAD para criar e desenvolver várias ideias relativas ao desenvolvimento do projeto.

Seguidamente foi realizada a fase de Seleção da Ideia, onde das várias ideias geradas e discutidas se optou por responder à oportunidade identificada com a implementação de um sistema de recomendação e componente de consulta de informação agregada.

Finalmente, foi realizada a Definição do Conceito onde foram determinados os vários objetivos a serem atingidos de modo a alcançar a solução pretendida, de uma forma que justifique a sua implementação ao cliente.

Com a realização destas cinco fases o processo pode passar para a fase de desenvolvimento do produto e finalmente a sua “comercialização” onde este é entregue ao cliente. Este processo é apresentado na Figura 15.

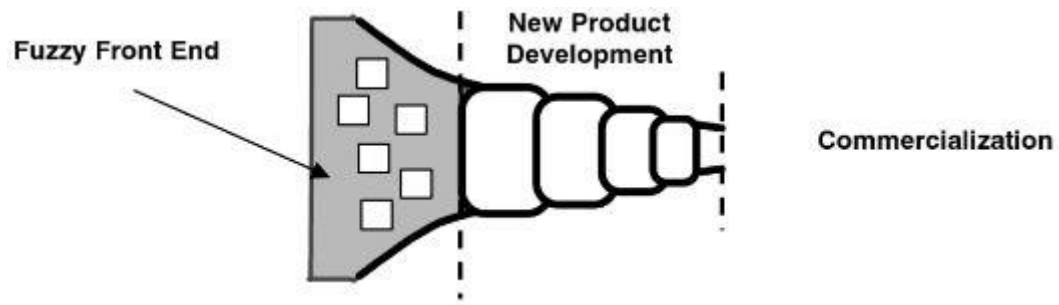


Figura 15 – Processo de Inovação (P. A. Koen et al. 2002)

### 2.6.2 Valor Percebido

O conceito de valor percebido pode ser descrito como sendo “a avaliação geral feita por um cliente face à utilidade de um produto (ou serviço) baseada no que é recebido e no que é dado” (Sweeney e Soutar 2001). O que implica que diferentes pessoas têm diferentes opiniões quanto ao valor do mesmo produto ou serviço.

No caso do valor percebido para a solução proposta, este é positivo visto pretende facilitar vários aspetos relativos ao processo de gestão de projetos, recursos humanos e consulta da sua informação, algo que é de grande interesse para o cliente.

Tendo falado do valor percebido da solução é abordado o valor da solução em si, que pode ser caracterizado através de benefícios e sacrifícios. Começa-se por indicar os benefícios:

1. Facilita ao cliente o processo de escolha de recursos humanos para equipas das várias candidaturas de projetos.
2. Disponibiliza uma forma mais simples e compreensível de consulta de informação de diferentes aspetos do negócio.
3. Estas funcionalidades são de fácil acesso, visto que fazem parte do sistema do GesProj.

No entanto em conjunto com os benefícios, também existem vários sacrifícios, que são apresentados de seguida:

1. O desenvolvimento da solução implica um investimento em termos de tempo e recursos por parte do GECAD.
2. A solução implica um aumento da complexidade do sistema já existente, pois envolve a alteração deste para integrar os componentes de recomendação e consulta de informação agregada.

3. O aumento da complexidade do sistema, implica uma maior dificuldade da manutenção deste como um todo.
4. Para o sistema de recomendação e componente de consulta de informação agregada funcionarem corretamente necessitam de quantidades consideráveis de informação digitalizada na base de dados do sistema.

### 2.6.3 Proposta de Valor

A proposta de valor, representa as diferentes ofertas que uma organização faz aos seus clientes, com o objetivo de ganhar o seu interesse, e fidelidade, face à sua competição (Osterwalder e Pigneur 2010). No caso do projeto tratado neste documento o cliente é a organização, ou seja, o GECAD, visto que o projeto pretende introduzir novas características num portal utilizado internamente.

Para estruturar e especificar a proposta de valor é utilizado o Canvas da Proposta de Valor, um modelo constituído pelo Mapa de Valor, que representa como se pretende criar valor, e Perfil do Cliente que representa as características do clientes (Osterwalder et al. 2014). A estrutura deste modelo é apresentada na Figura 16, com o Mapa de Valor à esquerda e Perfil do Cliente à direita.

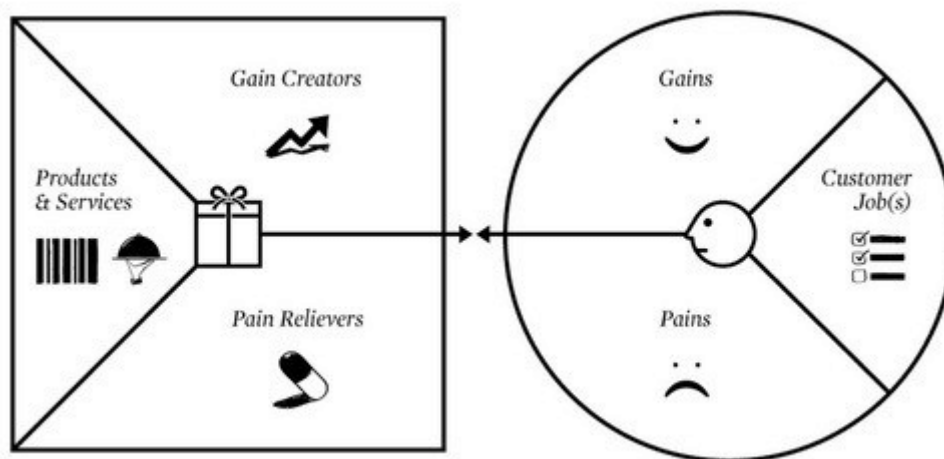


Figura 16 – Canvas da Proposta de Valor

Como é apresentado na Figura 16 o Perfil do Cliente é composto pelos seguintes componentes (Osterwalder et al. 2014):

- *Customer Job(s)*: O que o cliente pretende fazer.

- *Gains*: O que o cliente pretende alcançar.
- *Pains*: Os riscos e obstáculos relativos ao(s) *Customer Job(s)*.

Enquanto que o Mapa de Valor é composto maioritariamente pelas “respostas” às características dos clientes (Osterwalder et al. 2014):

- *Products & Services*: Os produtos e/ou serviços abordados pela proposta de valor.
- *Gain Creators*: Descreve como os *Products & Services* criam *Gains*.
- *Pain Relievers*: Descreve como os *Products & Services* aliviam *Pains*.

No caso do projeto a ser tratado nesta dissertação foram identificados os seguintes atributos para o Perfil do Cliente:

- *Customer Job(s)*: O GECAD pretende alocar recursos humanos a equipas para candidaturas a projetos. Em conjunto com consultar informação relativa aos projetos, recursos humanos, indicadores e objetivos dos vários projetos.
- *Gains*: O GECAD pretende ter o processo de alocação de recursos humanos a equipas facilitado. Também necessita de uma forma simplificada e mais acessível de consulta de informação.
- *Pains*: O GECAD gasta muito tempo na escolha dos diferentes elementos para as diferentes equipas. Atualmente realiza a gestão e consulta de informação maioritariamente através de folhas Excel que dificulta e complica esse processo.

No caso do Mapa de Valor foi realizada a seguinte análise:

- *Products & Services*: Sistema de recomendação de equipas e componente de consulta de informação agregada.
- *Gain Creators*: Através do sistema de recomendação é possível sugerir e escolher diferentes recursos humanos para as equipas das várias candidaturas de projetos em menos tempo. Com o componente de consulta de informação, a informação é apresentada numa forma estruturada e simples de aceder através de *dashboards*.
- *Pain Relievers*: O sistema de recomendação diminui o tempo necessário para a criação de equipas através das diferentes sugestões. Através do componente de consulta de informação, a informação é digitalizada e mantida no portal, removendo a necessidade da existência de várias folhas Excel para a gestão da mesma.

Através da utilização deste modelo é criada a seguinte **Proposta de valor**:

*A solução proposta apresenta uma alternativa aos processos atuais de criação de equipas para candidaturas a projetos, através do sistema de recomendação que facilita a escolha de diferentes elementos para as várias equipas. Em conjunto com o sistema de recomendação, é facilitada a consulta de informação, através de um componente de consulta de informação agregada, mais especificamente dashboards.*

A implementação do sistema de recomendação permite facilitar o processo de criação de equipas, visto que diminui o esforço de pensar e escolher os diferentes investigadores, pois o sistema é capaz de sugerir elementos com base em informação existente. Uma sugestão que pode ou não ser aceite, mas em qualquer dos casos apresenta um ponto de partida para a criação de uma nova equipa, retirando assim tempo à realização desta decisão que pode ser utilizado para outras tarefas.

No caso do componente de consulta de informação, este é uma alternativa mais fácil e organizada de utilizar que outras, por exemplo folhas de Excel (Anexo I), pois não só apresenta o aspeto de digitalização de informação devido ao facto desta se encontrar numa base de dados, mas também a estrutura de uma forma simples de compreender. Tal é possível através da criação de diferentes *dashboards* que apresentam a informação sobre a forma de diferentes tipos de gráficos.

#### **2.6.4 Quality function deployment (QFD)**

O método QFD, foi inicialmente desenvolvido no Japão entre o fim da década de 60 e início da de 70, com o objetivo de transformar os requisitos do cliente em requisitos funcionais a serem utilizados no desenvolvimento da solução. QFD é aplicado em várias áreas como por exemplo no área de software (Chan e Wu 2001).

Para aplicar o método QFD é utilizado um conceito denominado “Casa de Qualidade”, que tem informação “do que fazer”, “como o fazer” e a integração desta informação com o objetivo de determinar as várias características de engenharia (Kim et al. 2000). A estrutura da Casa a ser utilizada neste contexto é apresentada na Figura 17.

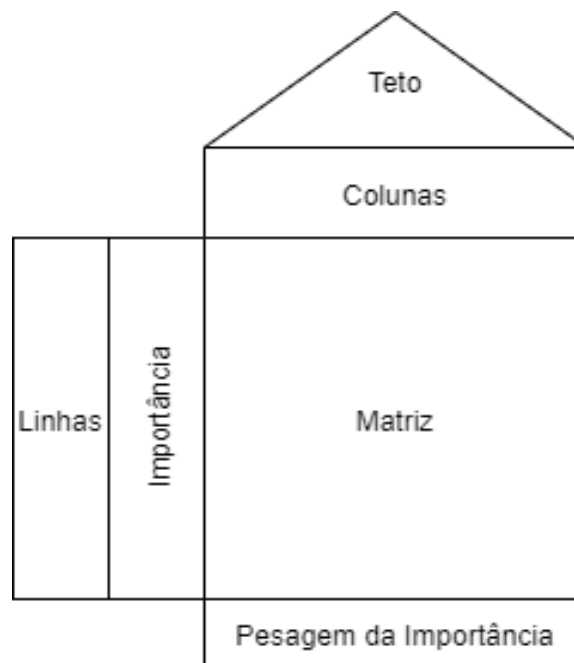


Figura 17 – Estrutura da Casa de Qualidade

Onde os elementos desta estrutura têm os seguintes significados (azizkhan 2012):

- **Teto:** A correlação entre os atributos presentes nas Colunas, através de símbolos;
- **Colunas ou “Funcionalidades da Solução”:** As funcionalidades do produto, que indicam como o produto pode satisfazer as necessidades do cliente;
- **Linhas ou “Necessidades do Cliente”:** As necessidades do cliente, ou seja, as características que estes acham importantes;
- **Importância:** A relevância que o cliente atribui aos requisitos numa escala de 1 a 6, onde 1 é o maior nível de significância e 6 o menor;
- **Matriz:** Contem símbolos que representam a relação entre requisitos e atributos, onde cada relação tem um valor de 1, 3 ou 9;
- **Pesagem da Importância:** Pesagem da importância relativa de cada característica do produto, através da soma dos símbolos contidos na Matriz, onde os maiores valores representam uma maior importância.

A simbologia utilizada para representar as relações no Teto e Matriz, encontram-se na Figura 18 e Figura 19 respetivamente.

Teto	
	Muito Positivo
	Positivo
	Negativo
	Fraco

Figura 18 – Símbolos do Teto

Matriz		
	Forte	9
	Médio	3
	Fraco	1

Figura 19 – Símbolos da Matriz

No contexto desta dissertação utilizou-se o método QFD analisar a relação entre as necessidades do cliente (GECAD) e como o produto (solução a ser desenvolvida) é capaz do proporcionar. Para tal foram estabelecidas as seguintes funcionalidades do produto ou Colunas:

- Sistema de Recomendação de recursos humanos para a criação de equipas;
- Componente de consulta de informação agregada que, através de *dashboards*, permite a consulta do ponto de situação dos vários projetos em curso num determinado momento;
- Atualização e integração dos sistemas com o GesProj.

Em conjunto com as seguintes necessidades do cliente ou Linhas:

- Recomendação de investigadores com base no seu perfil científico;
- Escolha de investigadores com base em recomendações;
- Consulta da percentagem de afetação de investigadores;
- Consulta do ponto de situação de projetos;
- Consulta de indicadores de realização de projetos.

Sendo assim, foi criada a Casa de Qualidade apresentada na Figura 20.

Funcionalidades da Solução Necessidades do Cliente	Relevância para o Cliente	Sistema de Recomendação de recursos hum anos	Componente de consulta de informação que permite a consulta do estado de situação do vários projetos	Atualização e integração dos sistemas com o GesProj
Recomendação de investigadores com base no seu perfil científico	1	●		●
Escolha de investigadores com base em recomendações	2	●		●
Consulta da percentagem de afetação de investigadores	1		●	●
Consulta do ponto de situação de projetos	1		●	●
Consulta de indicadores de realização de projetos	3		●	●
Pesagem da Importância		12	9	15

Figura 20 – Casa de Qualidade

### 2.6.5 Modelo Canvas

O Modelo Canvas é uma ferramenta utilizada para descrever, analisar e desenhar elementos do modelo de negócio (Osterwalder e Pigneur 2010). O modelo desenvolvido para esta dissertação é apresentado na Figura 21.

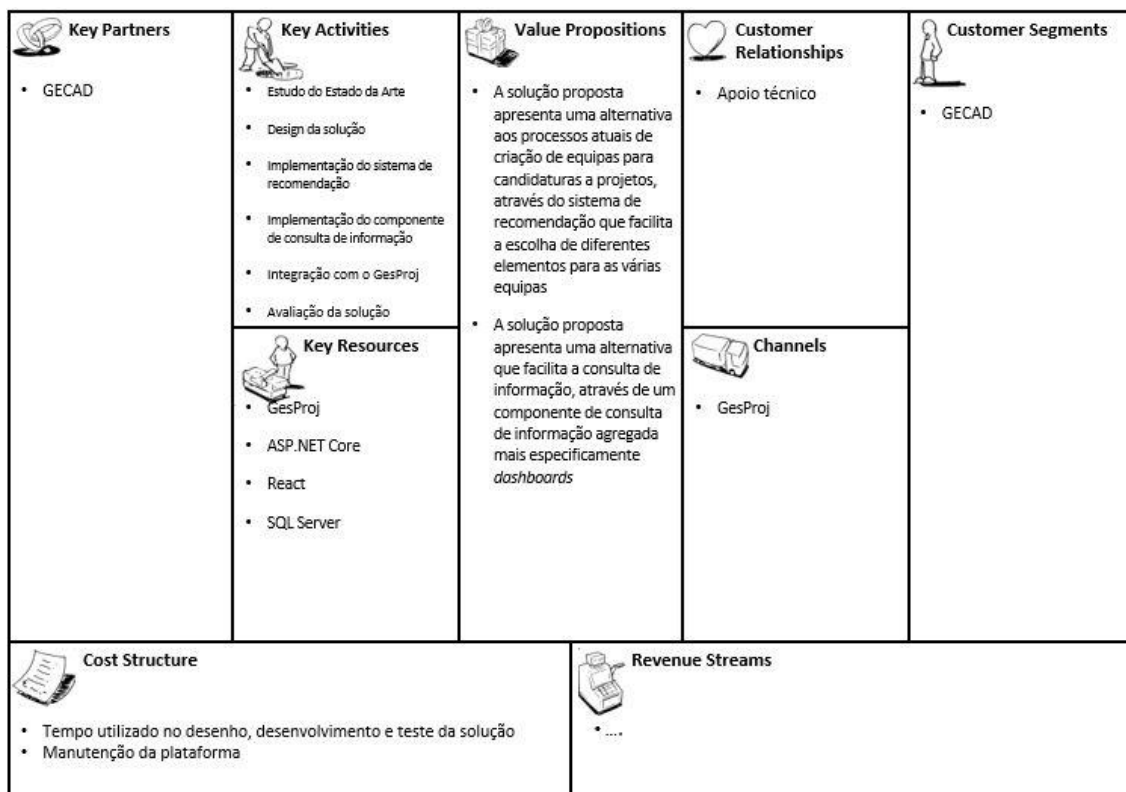


Figura 21 – Modelo Canvas

No caso dos Parceiros Chave (*Key Partners*) e Segmentos dos Consumidores (*Customer Segments*) estes correspondem ao GECAD, visto que o projeto é desenvolvido para o GECAD com o apoio do mesmo.

A Proposta de Valor (*Value Propositions*) já foi referida na respetiva secção Proposta de Valor, mas é aqui representada mais uma vez no contexto do Modelo Canvas.

As Atividades Chave (*Key Activities*) incluem o estudo do Estado da Arte, design da solução, implementação do sistema de recomendação, e do componente de consulta de informação, seguidos da integração com o portal do GECAD e finalmente da avaliação da solução. Para a realização destas atividades são necessários os Recursos Chave (*Key Resources*), o portal GECAD em si pois é a base do projeto e as várias tecnologias envolvidas (*ASP.NET Core*, *React* e *SQL Server*).

As Relações com o Cliente (*Customer Relationships*) são representadas pelo apoio técnico disponibilizado durante o desenvolvimento, implantação e compreensão da solução. Enquanto que os Canais (*Channels*) são representados pelo portal em si, visto que é através deste que se interage com os diferentes sistemas.

A Estrutura de Custos (*Cost Structure*) inclui o tempo despendido no desenvolvimento e teste dos diferentes sistemas e a manutenção da plataforma.

## 3 Design

Neste capítulo, é descrita a arquitetura da solução proposta como resposta ao problema apresentado por esta dissertação. São apresentados os requisitos do problema e é definida a arquitetura da solução a partir de duas possibilidades. Também é realizada uma análise do domínio e de processos que permitem perceber o funcionamento do sistema.

### 3.1 Requisitos

O design de uma solução é afetado por diferentes requisitos que precisam de ser identificados, para identificar os requisitos da solução ao problema abordado neste documento é utilizado o modelo FURPS+.

O modelo FURPS+ apresenta um conjunto de diferentes categorias onde os vários requisitos podem ser colocados (Eeles 2001). Sendo estas:

- *Functionality* ou Funcionalidade;
- *Usability* ou Usabilidade;
- *Reliability* ou Confiabilidade;
- *Performance* ou Desempenho;
- *Supportability* ou Suporte;
- O “+” é utilizado para representar restrições extra, podendo ser aplicadas sobre o design, a implementação, a interface e aspetos físicos.

#### 3.1.1 Requisitos Funcionais

Os requisitos funcionais, representados pelo “F” do modelo FURPS+, correspondem às principais funcionalidades e características que o cliente pretende ter na solução (Eeles 2001).

Para melhor especificar as funcionalidades do sistema e como estas se integram no processo de gestão de projetos, foram identificados dois utilizadores:

- **Administrador:** O utilizador que tem acesso a todas a funcionalidades disponibilizadas pelo sistema.
- **Gestor de Projeto:** Representa um utilizador que tem a responsabilidade de gerir a execução de um projeto, logo consegue gerar recomendações para as suas equipas,

consultar a informação relativa a projetos, afetações, e indicadores e alterar informação relativa a projetos.

No entanto, como esta dissertação não se foca na implementação do sistema de login ou papéis dentro do sistema, estas designações têm apenas o papel de melhor diferenciar as várias funcionalidades e quem as pode utilizar do ponto de vista do negócio.

Para o problema em questão foram especificados os seguintes requisitos e funcionalidades:

- Sistema de Recomendação de recursos humanos para formação de equipas para projetos de investigação
  1. O sistema deve permitir ao administrador ou gestor de projeto gerar recomendações de recursos humanos com base nas diferentes *tags* que caracterizam as atividades envolvidas na realização de um projeto, período de realização do projeto, número de investigadores que pretende ter na equipa e limite de afetação desses investigadores.
- Consulta de informação através do acesso a *dashboards* que permitam a consulta do ponto de situação dos vários projetos em curso na organização num determinado momento
  1. O sistema deve permitir ao administrador ou gestor de projeto consultar as percentagens de afetação de investigadores a projetos com a capacidade de controlar a granularidade da consulta (todos os projetos ou apenas alguns). Esta consulta deve ser feita através de um gráfico e tabela.
  2. O sistema deve permitir ao administrador ou gestor de projeto consultar uma *timeline* de um projeto, que apresente um *feedback* do seu ponto de situação, alertando caso haja situações irregulares. Esta consulta deve ser feita através de um gráfico.
  3. O sistema deve permitir ao administrador ou gestor de projeto consultar os diferentes indicadores realizados e por realizar do projeto. Esta consulta deve ser feita através de um gráfico e tabela.
- Gestão de Áreas Científicas
  1. O sistema deve permitir ao administrador introduzir, eliminar e editar áreas científicas. Uma área científica é constituída pelo nome.
- Gestão de *Milestones*

1. O sistema deve permitir ao administrador ou gestor de projeto introduzir, eliminar e editar *milestones* em relação a uma atividade. Uma *milestone* é constituída pelo nome, data limite de realização, estado (por realizar ou realizada) e observações quanto à sua realização.
- Gestão de Indicadores Realizados
    1. O sistema deve permitir ao administrador ou gestor de projeto introduzir, eliminar e editar indicadores realizados em relação a um indicador a realizar. Um indicador realizado é constituído pelo valor realizado, data de realização, observações e opcionalmente a *milestone* com qual esta relacionado.
  - Gestão de Investigadores
    1. O sistema deve permitir ao administrador, quando este utilizada a funcionalidade de criação/edição de investigadores do GesProj, indicar quais as áreas em que este tem competências.
  - Registo de Atividades
    1. O sistema deve permitir ao administrador ou gestor de projeto, quando este utilizada a funcionalidade de criação/edição de atividades do GesProj, indicar qual a área (competência) necessária para a realizar.

Desta análise resultam os diferentes casos de uso que o cliente espera serem disponibilizados pela solução, apresentados na Figura 22.



Figura 22 – Diagrama de Casos de Uso

### 3.1.2 Requisitos Não Funcionais

Os requisitos não funcionais, representados pelo “URPS+” do modelo FURPS+, correspondem às características com implicações arquitetónicas ou que limitam certos aspetos do processo de desenvolvimento (Eeles 2001). Para o problema em questão foram especificados os seguintes requisitos:

- Usabilidade
  1. A interface do utilizador com o componente de consulta de informação agregada (*dashboards*) deve apresentar informação de uma forma simples de aceder e compreender.
- Suporte
  1. O sistema como um todo deve guardar os detalhes de um erro num log, mas apenas apresentar uma mensagem de erro genérica.
  2. O sistema como um todo deve ser de fácil manutenção.
- Restrições de design

1. Tanto o componente responsável por recomendações e pela consulta de informação devem ter acesso ao componente de base de dados do GesProj (SQLServerDb).
- Restrições de implementação
    1. Aplicações em *back-end* devem ser implementadas utilizando *ASP.NET Core*.
    2. Aplicações em *front-end* devem ser implementadas utilizando *React*.
  - Restrições de interface
    1. Ambos os novos componentes devem ser acedidos via browser.

## 3.2 Arquitetura

Seguidamente são apresentadas diferentes propostas de arquiteturas que podem ser adotadas como solução, e destas é escolhida uma. Visto que o sistema de recomendação e componente de consulta de informação tem de integrar com o GesProj, qualquer arquitetura discutida tem de conter os componentes presentes na Figura 3. A nova solução também deve manter a abordagem modular da solução inicial, pois esta facilita a manutenção da solução e dos seus componentes (como é pedido nos Requisitos Não Funcionais), a depuração de erros e promove um baixo acoplamento.

Na Figura 23 é apresentada a primeira abordagem que implica a criação de dois novos componentes, que interagem diretamente com a estrutura já existente do GesProj.

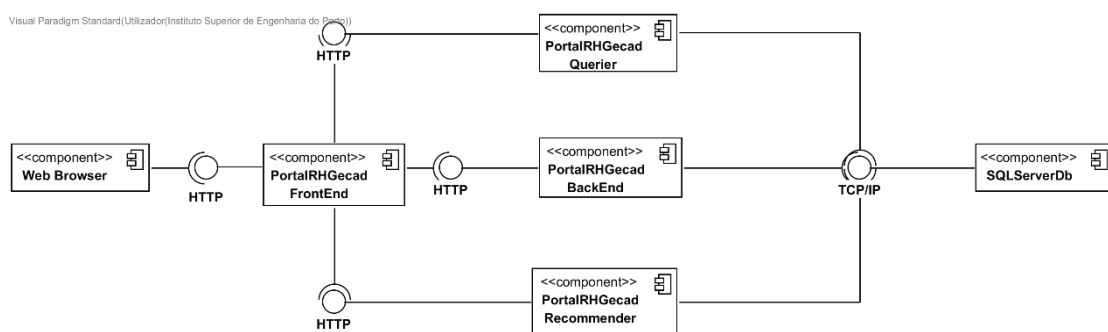


Figura 23 – Arquitetura 1

No caso desta proposta de arquitetura o componente Web Browser mantém-se inalterado, enquanto que o componente PortalRHGecadFrontEnd será alterado para se conseguir ligar às três aplicações servidor e depois apresentar as funcionalidades disponibilizadas por estas no Web Browser. No caso dos componentes SQLServerDb e PortalRHGecadBackEnd estes serão

alterados para acomodar a criação de novos conceitos, e novas funcionalidades, em conjunto com a atualização de conceitos e funcionalidades já existentes.

O componente PortalRHGecadRecommender será responsável por, através da sua ligação à base de dados e aplicação cliente do GesProj, criar as diferentes recomendações desejadas pelos utilizadores. O componente PortalRHGecadQuerier será responsável por através da sua ligação com o componente SQLServerDb estruturar informação para esta depois ser apresentada no Web Browser através do componente PortalRHGecadFrontEnd.

Na Figura 24 é apresentada a segunda abordagem que implica a criação de três novos componentes, onde um destes é responsável pela maioria da integração dos outros dois com o GesProj.

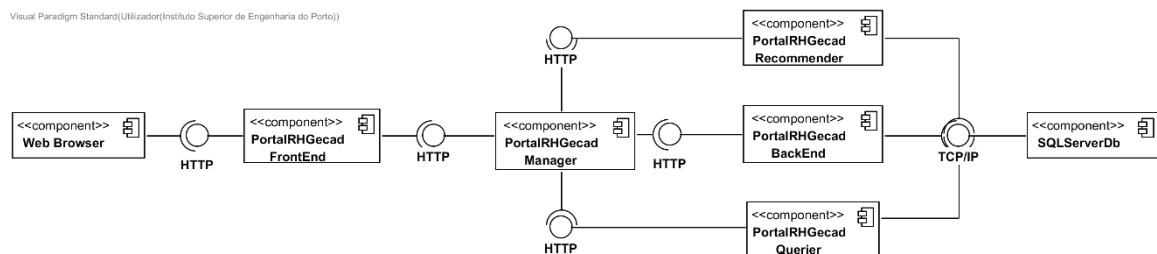


Figura 24 – Arquitetura 2

No caso desta proposta de arquitetura o componente Web Browser mantém-se inalterado, enquanto que o componente PortalRHGecadFrontEnd será alterado para se conectar ao componente PortalRHGecadManager e depois apresentar as funcionalidades disponibilizadas por este no Web Browser. Os componentes SQLServerDb e PortalRHGecadBackEnd serão atualizados para acomodar as necessárias alterações no domínio e funcionalidades.

O componente PortalRHGecadRecommender será responsável por, através da sua ligação ao componente SQLServerDb e PortalRHGecadManager, criar as diferentes recomendações desejadas pelo cliente. O componente PortalRHGecadQuerier será responsável por, através da sua ligação com o componente SQLServerDb, estruturar informação para depois o componente PortalRHGecadManager a disponibilizar ao PortalRHGecadFrontEnd.

O componente PortalRHGecadManager funcionaria como uma API Gateway, visto que será responsável por tornar transparente a comunicação entre o PortalRHGecadFrontEnd e as três aplicações servidor.

Comparando as duas propostas, estas apenas se distinguem pela presença do componente PortalRHGecadManager na segunda proposta.

O componente PortalRHGecadManager é responsável por tornar transparente a comunicação entre a aplicação cliente e aplicações servidor e diminuir a quantidade de alterações que necessitam de ser feitas no componente PortalRHGecadFrontEnd. No entanto esta diminuição de alterações é mínima, especialmente quando o componente necessita de ser alterado para conseguir disponibilizar ao utilizador todas as novas funcionalidades.

É preciso ainda considerar que a segunda abordagem aumenta a complexidade da solução pois implica a criação e manutenção de um novo componente. Também se deve considerar o facto de que a solução final pertence ao GECAD (cliente), logo este controla a implantação e estado dos componentes da solução. Não havendo assim tantas preocupações relativamente ao funcionamento correto dos serviços, relativamente ao formato da informação disponibilizada por estes ou correta integração de diferentes componentes da solução.

Sendo assim não existe uma boa justificação para a adoção da arquitetura representada pela Figura 24, escolhendo-se assim a primeira opção representada pela Figura 23.

### 3.3 Conceitos do Domínio

Nesta secção são abordados os vários conceitos de negócio envolvidos no problema abordado neste documento.

Primeiro é necessário mencionar que a maioria dos conceitos do domínio mantêm-se face ao que foi apresentado na secção GesProj, mais especificamente na Tabela 1 e Figura 1. No entanto é necessário alterar alguns destes conceitos para permitir a implementação de um sistema de recomendação, componente de consulta de informação agregada e complementar alguns dos conceitos já existentes. Nesta secção começa-se por abordar os conceitos que complementam os conceitos já existentes, seguidos dos conceitos necessários para a implementação dos novos componentes.

No domínio apresentado na Tabela 1, existem indicadores base (*Indicator*) e indicadores a realizar (*IndicatorQuantified*), no entanto não existe um conceito que registe os indicadores realizados. No mesmo domínio também são consideradas as várias atividades de um projeto (*Activity*), mas não se consideram os diferentes objetivos que devem ser atingidos na realização

destas atividades. De modo a cobrir estas falhas no domínio foram criados os conceitos de indicador realizado (*IndicatorRealized*) e *Milestone*, que são apresentados na Tabela 12.

O conceito de Projeto (*Project*) necessita de ser atualizado face aos novos componentes do sistema, tendo de agora se considerar a sua data de início e fim, em conjunto com a indicação de se este se encontra concluído ou não.

Um conceito importante que necessita de ser melhor explicado, pois é abordado pelo sistema de recomendação e consulta de informação, é o de afetação. Quando um investigador é associado a um projeto, este é afeto ao projeto e esta percentagem de afetação é representada por um valor numérico. Este valor é estabelecido para o período de duração do projeto e depois dividido por cada mês desse período, por exemplo se o projeto durar dois meses e o investigador é afetado com um valor de 0.5, este tem uma percentagem de 0.25 em cada mês.

O sistema de recomendação pretende sugerir recursos humanos para a integração destes em equipas de projetos de investigação, para isso ser possível é necessário encontrar uma forma de ligar investigadores a atividades de um projeto. Uma primeira abordagem é através das afetações, pois permite saber quais as atividades em que um investigador está, ou esteve, a participar, no entanto isto não é suficiente para inferir se um investigador pode ou não ser afetado a uma dada atividade. Isto deve-se ao facto de mesmo que um investigador tenha estado relacionado com uma atividade, não existe informação concreta que este tem as competências para a realizar corretamente.

Para resolver este problema foram criados os conceitos de área (*Subject*), *Tag*, perfil científico (*ScientificProfile*) e Candidatura (*Application*), como é apresentado na Tabela 12. Enquanto que a Candidatura não ajuda no relacionamento entre investigadores e áreas, esta estabelece a informação de um projeto a ser considerada pelo sistema de recomendação.

Tabela 12 – Glossário de novos conceitos

<b>Termo</b>	<b>Significado</b>
Área ( <i>Subject</i> )	Área científica
Candidatura ( <i>Application</i> )	Representa a fase inicial de um Projeto antes deste ser inserido no sistema, onde são definidos os seus diferentes aspetos, como as suas atividades, indicadores e <i>milestones</i>
Indicador Realizado ( <i>IndicatorRealized</i> )	Um indicador realizado durante a duração de um Projeto, pode estar relacionado a uma <i>Milestone</i>
Perfil Científico ( <i>ScientificProfile</i> )	As diferentes Áreas onde um Investigador tem competências
<i>Tag</i>	A Área que identifica a competência envolvida na realização de uma Atividade
<i>Milestone</i>	Um objetivo a ser atingido na realização de uma Atividade

Através destes conceitos já é possível relacionar investigadores e atividades através das áreas científicas que têm em comum, e assim criar recomendações com base nas *tags* das atividades de um projeto e nos perfis dos vários investigadores.

As ligações entre os diferentes conceitos do domínio são apresentadas seguidamente no modelo de domínio atualizado do GesProj na Figura 25.

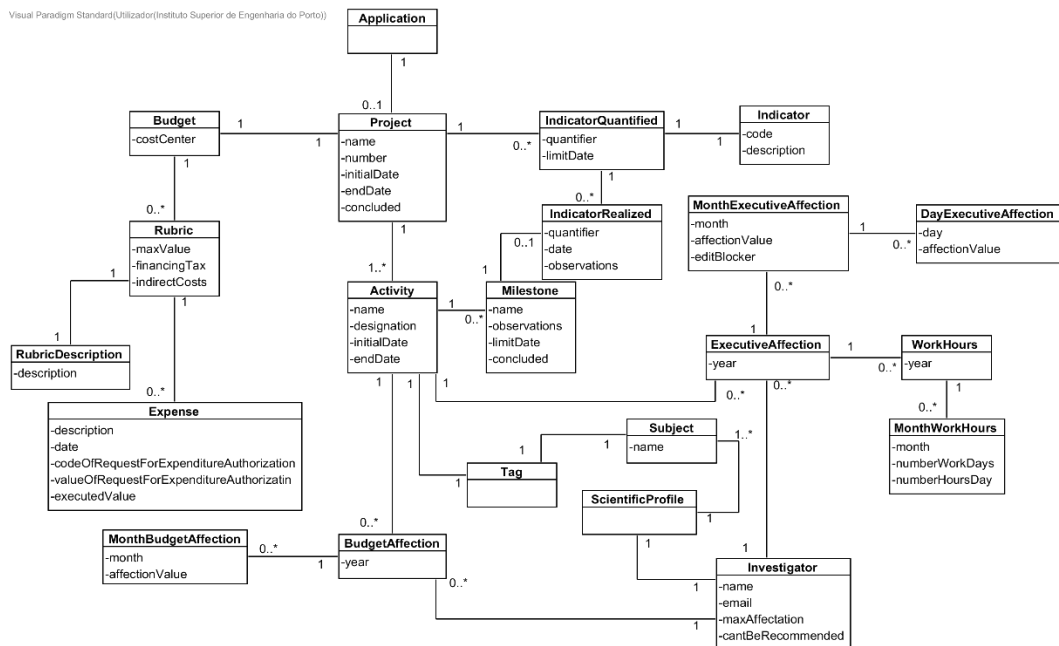


Figura 25 – Modelo de Domínio

Tendo criado o modelo de domínio atualizado, este agora é comparado com o modelo original apresentado na Figura 1. A diferença mais óbvia nestes dois modelos é a presença dos novos conceitos apresentados nesta secção, que são apresentados no modelo com os atributos especificados na secção de Requisitos Funcionais. Em conjunto com as novas entidades, também foram alterados ou adicionados atributos a entidades já existentes.

No caso da entidade *Project*, esta agora tem os atributos *initialDate*, *endDate* e *concluded*, para representar as datas de início e fim do Projeto, e o estado do Projeto quanto ao facto de este estar concluído. A entidade *Activity*, já não tem o atributo *attribute*, pois este agora é representado pela entidade *Tag*. No caso da entidade *Investigator*, esta perde os atributos *projectsAffection* e *totalAffection*, mas tem lhe adicionada os atributos *maxAffection* e *cantBeRecommended*, que representam a afetação máxima possível para o Investigador e um indicador de se este pode ou não ser considerado para recomendações. A entidade *IndicatorQuantified* é atualizada com o novo atributo *limitDate*, que representa a data limite até qual o indicador pode ser realizado.

### 3.4 Base de Dados

Com base na análise realizada na secção Conceitos do Domínio e o resultante modelo de domínio presente na Figura 25, foi criado o modelo presente na Figura 26 para representar a

base de dados a ser utilizada pelo GesProj para os novos sistemas funcionarem. No entanto, como é visível na Figura 26, a transição de modelo de domínio para modelo da base de dados não é transparente.

No caso da relação entre *Milestone* e *IndicatorRealized*, optou-se por criar a tabela intermedia *MilestoneIndicators*, devido ao facto de no domínio ser especificado que um Indicador Realizado pode ou não estar relacionado com uma *Milestone*. Dai optou-se por criar uma tabela intermedia entre as duas tabelas para representar esta relação, em vez de uma coluna “*MilestoneId*” na tabela *IndicatorRealized* que pudesse ser ou não *null*.

Quanto ao conceito de Perfil Científico, visto pode ser definido como a lista de Áreas (tabela *Subject*) onde um Investigador (tabela *Investigator*) tem competência, optou-se por utilizar a tabela *ScientificProfile* como uma tabela intermédia. Sobre a forma de uma tabela intermédia, a tabela *ScientificProfile* é capaz de representar as diferentes relações que um Investigador tem com as várias Áreas. De um ponto de vista de implementação esta decisão não afeta como o código pode vir a representar o domínio.

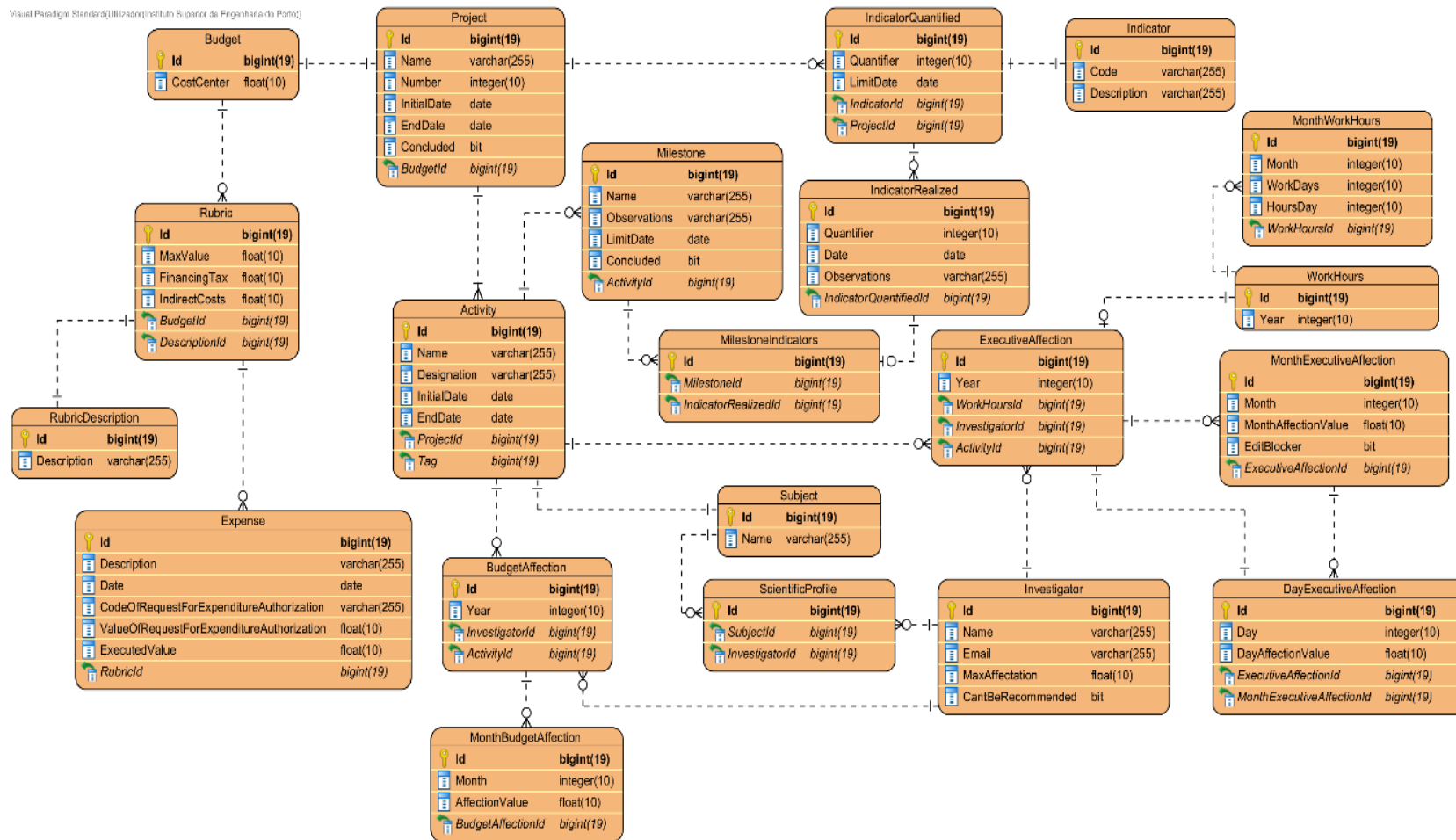


Figura 26 – Modelo da Base de Dados

## 3.5 Processos

Nesta secção são abordados alguns dos casos de uso mencionados na secção Requisitos Funcionais numa perspetiva de alto nível, para demonstrar como os diferentes componentes do sistema interagem entre si e a lógica mais importante do seu funcionamento.

### 3.5.1 Recomendação de recursos humanos

Começa-se por apresentar o funcionamento da recomendação de recursos humanos para uma candidatura de um projeto, através da Figura 27.

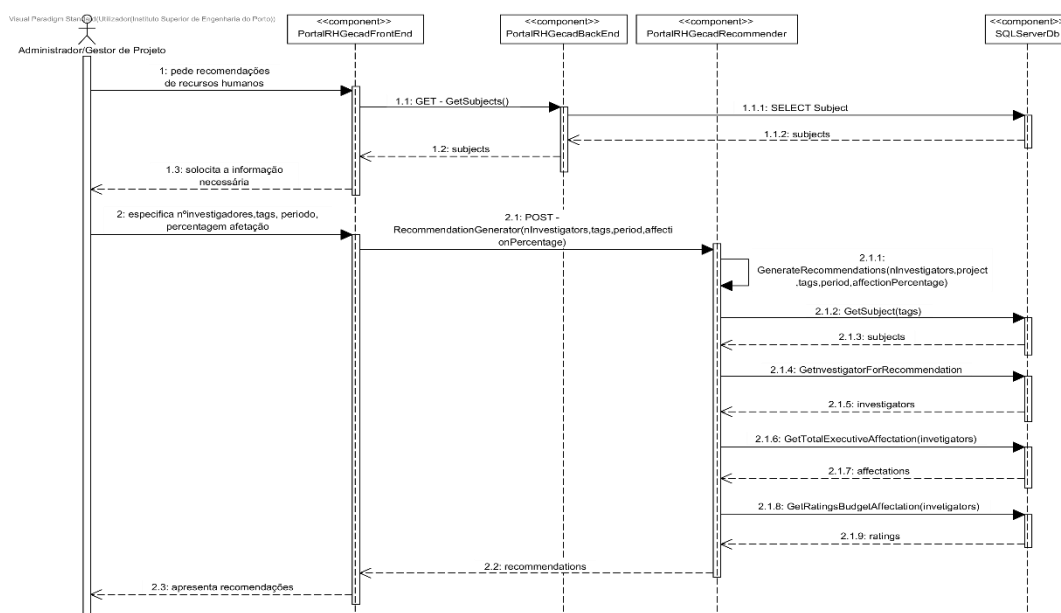


Figura 27 – Diagrama de sequência da recomendação de recursos humanos

Enquanto que esta funcionalidade está sempre disponível para ser utilizada, esta destina-se apenas a ser utilizada para a criação de equipas para candidaturas de projetos. Ou seja, antes do utilizador gerar recomendações necessita de ter acesso a uma candidatura de um projeto, e através desta determinar quais as *tags* (áreas científicas) envolvidas na sua realização, o período durante o qual o projeto da candidatura irá decorrer. Ao mesmo tempo necessita de determinar quantos investigadores pretende alocar à equipa e para todos estes qual a afetação total que estes podem atingir quando afetados a este projeto.

Com esta informação definida o utilizador indica na página *web* as *tags* que necessitam de recursos, o número de recursos (investigadores) necessários, o período de tempo que se

pretende preencher (data de início e fim) e a percentagem de afetação limite para os recursos a serem sugeridos.

Através de um pedido HTTP POST à aplicação servidor, são feitos vários pedidos à base de dados para esta depois ser utilizada na geração de recomendações que no fim são devolvidas ao utilizador para este consultar.

### 3.5.2 Consulta de percentagens de afetação dos investigadores aos projetos

Começa-se por apresentar a consulta de afetações de investigadores a projetos, através da Figura 28.

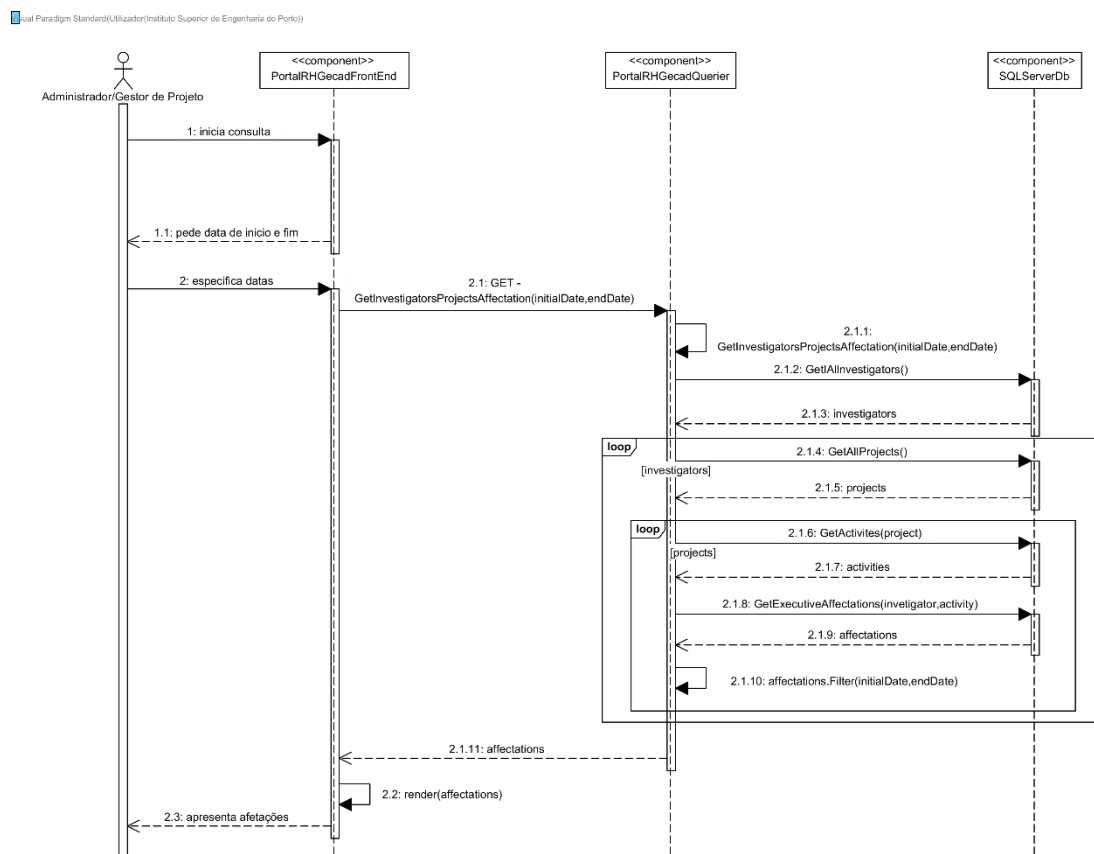


Figura 28 – Diagrama de sequência da consulta de afetações aos projetos

Quando o utilizador pretende consultar as percentagens de afetação dos investigadores aos projetos, necessita de escolher o período a considerar com a especificação de uma data de início e fim. Após realizar a escolha é realizado um pedido HTTP GET à aplicação servidor, que realiza as necessárias consultas à base de dados para obter a informação necessária para determinar as percentagens de afetação dos vários investigadores no período indicado.

Com esta informação a aplicação consegue formatá-la e enviá-la de volta à aplicação cliente para ser apresentada na *dashboard*. No caso do gráfico é espectável que este apresenta a opção de filtrar as afetações de diferentes investigadores.

No entanto, no caso da tabela, esta filtragem de informação deverá ser realizada através da realização de pedidos HTTP com diferentes tipos de parâmetro. Implicando para lá da especificação da data de início e fim da consulta, também deve ser possível indicar o investigador e/ou projeto (através do seu id da base de dados) para a realização de consulta. A especificação destes parâmetros não afeta o *design* apresentado na Figura 28, apenas implicando a possibilidade de especificar o investigador e/ou projeto no *front-end*.

### 3.5.3 Consulta da *timeline* de um projeto

Começa-se por apresentar o funcionamento da consulta da *timeline* de um projeto, através da Figura 29.

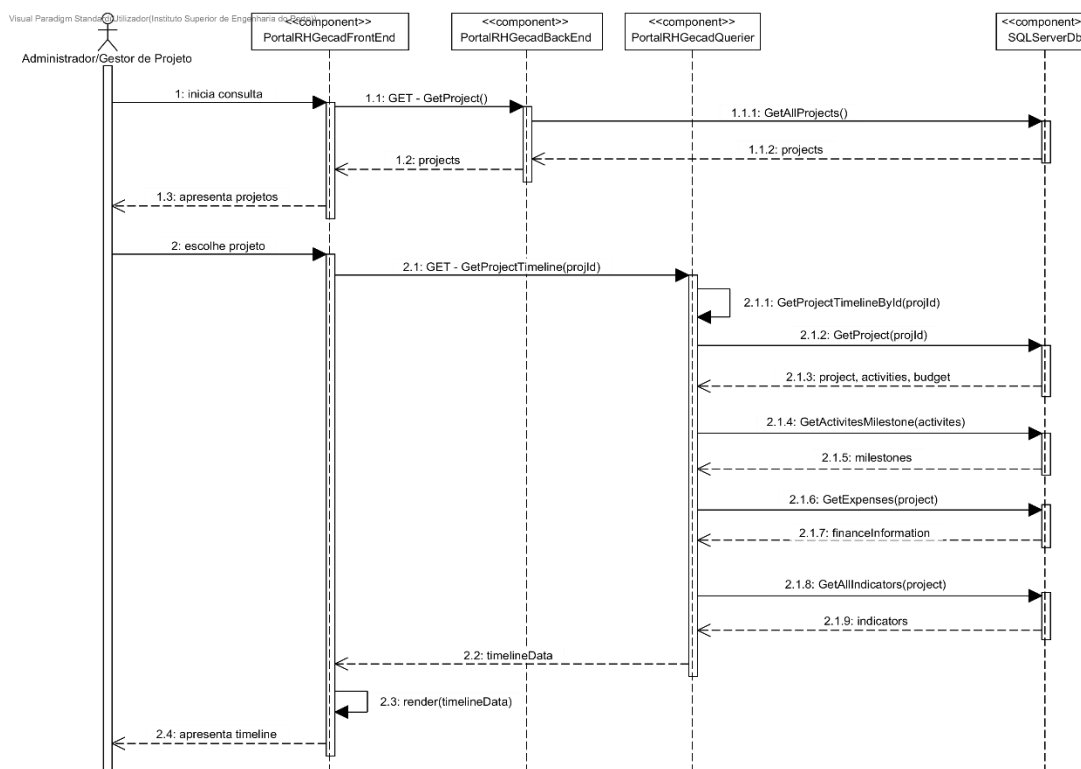


Figura 29 – Diagrama de sequência da consulta da *timeline* de um projeto

Primeiro é necessário mencionar o que se entende como *timeline* do ponto de vista deste caso de uso. Neste contexto uma *timeline* refere-se a uma representação gráfica do fluxo de execução do projeto, que começa na sua data de início e termina na sua data de fim, e durante

este período são apresentadas as suas atividades e o conjunto de eventos que ocorreram (por exemplo pedidos de pagamento) e eventos a decorrer (por exemplo indicadores por realizar). Também deve ser apresentado o ponto de situação do projeto como um todo, em conjunto com alguma situação irregular, como por exemplo incumprimento de indicadores, *milestones* ainda não atingidas depois da sua data limite, entre outras.

O utilizador quando pretende consultar a *timeline* de um projeto, necessita de primeiro escolher um projeto, através de uma lista recebida por um pedido GET. Após realizar a escolha é realizado outro pedido HTTP GET à aplicação servidor, que realiza as necessárias consultas à base de dados para obter a informação necessária relativa ao projeto.

Com esta informação a aplicação consegue formatá-la a enviá-la de volta à aplicação cliente para ser apresentada na *dashboard* com a *timeline* do projeto.

### 3.5.4 Consulta de indicadores por projeto

Começa-se por apresentar o funcionamento da consulta de indicadores por projeto, através da Figura 30.

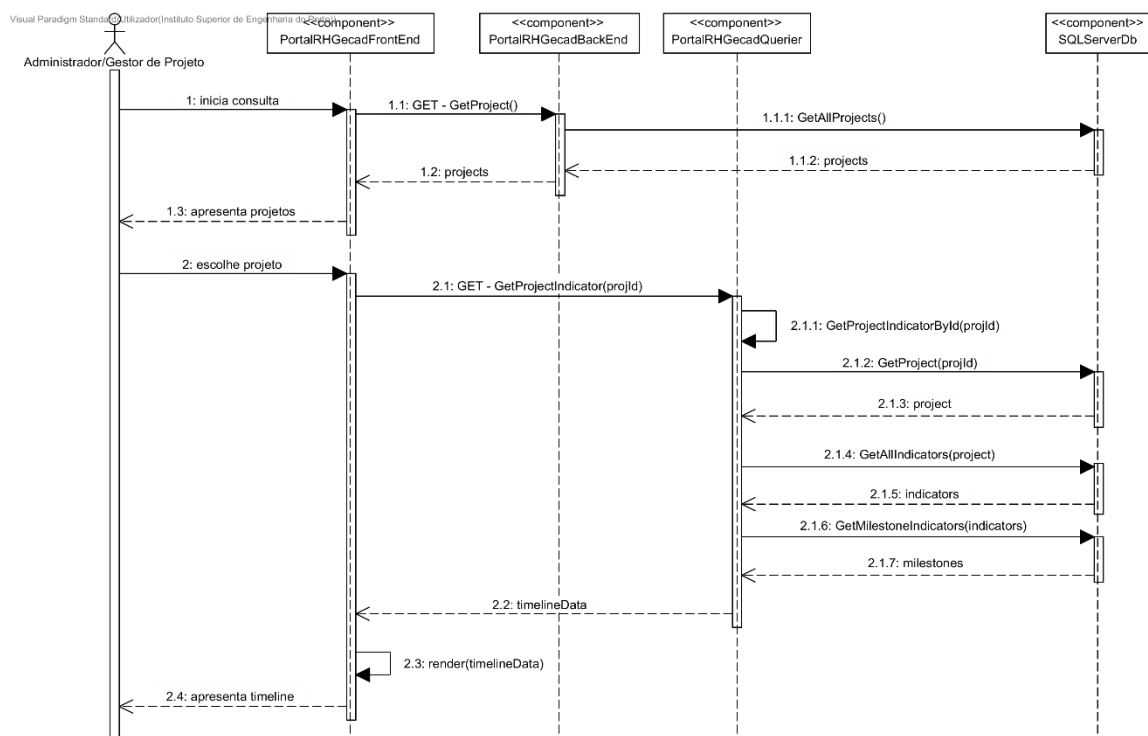


Figura 30 – Diagrama de consulta de indicadores por projeto

Tal como na consulta da *timeline* de um projeto, primeiro o utilizador tem de escolher o projeto de quais os indicadores pretende consultar. Com o projeto escolhido é realizado HTTP GET à

aplicação servidor, que faz várias consultas à base de dados para obter a informação necessária relativa aos indicadores, que depois é formatada para ser enviada à aplicação cliente.



## 4 Desenvolvimento

Neste capítulo, é apresentado o processo de implementação dos componentes e funcionalidades propostas no capítulo de Design. Esta secção é assim dividida pelos diferentes componentes apresentados na Arquitetura.

### 4.1 SQLServerDb

Este componente mantém a sua implementação em *SQL Server* tal como no GesProj, no entanto a estrutura da base de dados foi alterada com base no modelo que foi apresentado na secção de Base de Dados.

Quanto à implementação da estrutura da base de dados, esta foi feita através do sistema de migrações disponibilizado pelo *Entity Framework Core* (bricelam 2018). Que permitiu utilizar as entidades do modelo do PortalRHGecadBackEnd, para criar a estrutura da base de dados e posteriormente atualizá-la conforme a evolução destas entidades durante o processo de desenvolvimento.

Para ter acesso direto à base de dados e à sua informação foi utilizado o *SQL Management Studio* da Microsoft.

### 4.2 PortalRHGecadBackEnd

No caso do componente em questão, este já existia como parte do GesProj. No entanto necessitava de alterações e atualizações, com base nos Conceitos do Domínio, para acomodar novos conceitos e permitir o seu funcionamento em conjunto com os novos componentes a serem desenvolvidos.

Para facilitar a explicação do processo de desenvolvimento é apresentado na Figura 31 a estrutura de diretórios do projeto.

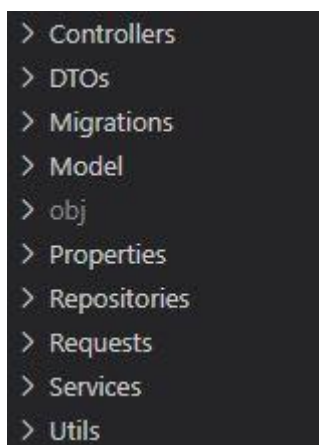


Figura 31 – Estrutura de diretórios do PortalRHGecadBackEnd

O diretório *Model* contém as classes representantes das entidades do negócio. O diretório *Controllers* contém os diferentes *controllers* responsáveis por receber e tratar os vários pedidos HTTP. As funcionalidades necessárias para permitir o processamento dos pedidos estão presentes nos serviços contidos no *Services*. Estes depois são capazes de permitir a gestão das diferentes entidades do negócio com a utilização dos repositórios, presentes no *Repositories* e das funcionalidades utilitárias, nos *Utils*. Devolvendo no fim ao *controller* um *Data Transfer Object* (DTO) que é utilizado pelo *controller* para criar a resposta ao pedido.

Sendo assim, para criar as entidades representantes dos conceitos apresentados na Tabela 12, mais especificamente os conceitos de *IndicatorRealized*, *Milestone*, *Subject*, *Tag* e *ScientificProfile*, foi utilizada esta estrutura. É de seguida exemplificado o processo de implementação com base na entidade *Subject* e na funcionalidade da consulta de uma Área com base no seu id.

#### 4.2.1 Consulta de uma Área com base no seu id

Primeiro tem de ser criada a classe representante da entidade *Subject*, no diretório *Model* como é apresentado na Figura 32.

```

namespace portalRHGecad.Model
{
    /// <summary>
    /// Subject associated with an activity or scientific profile
    /// </summary>
    0 references
    public class Subject
    {
        0 references
        public long Id { get; set; }

        /// <summary>
        /// Subject name
        /// </summary>
        [Required]
        0 references
        public string Name { get; set; }
    }
}

```

Figura 32 – Classe Subject

Tendo criado a classe, é necessário criar o respetivo repositório, serviço e *controller*, neste caso o código apresentado irá focar-se nos métodos e classes que permitem a consulta de uma área com base no seu id. Primeiro foi implementado o repositório que permite o acesso à base de dados, como é apresentado na Figura 33 e Figura 34.

```

public async Task<Subject> GetSubjectByIdAsync(long id)
{
    return await GetByIdAsync(id);
}

```

Figura 33 – Método da classe SubjectRepository

```

public async Task<T> GetByIdAsync(long id)
{
    return await this.context.Set<T>().FindAsync(id);
}

```

Figura 34 – Método base de consulta com id

Através da invocação do método *GetByIdAsync* é realizada uma consulta que devolve a Área cujo id é igual ao recebido. O acesso aos vários métodos disponibilizados pelo repositório, é realizado através de um serviço, como o que é exemplificado na Figura 35.

```

public async Task<SubjectDto> GetSubjectById(long subjectId)
{
    var subject = await subjectRepository.GetSubjectByIdAsync(subjectId);

    if (subject == null)
    {
        return null;
    }

    return mapper.Map<SubjectDto>(subject);
}

```

Figura 35 – Método da classe *SubjectService*

No método *GetSubjectById* é adquirida a Área através do seu id com o método *GetSubjectByIdAsync*, no entanto para este método ser utilizado pelo *controller* tem de ainda transformar o objeto no seu respetivo DTO. Para realizar a transformação de entidade para DTO é utilizado um *automapper*, que consegue fazer o mapeamento automaticamente através de um perfil. Perfil este que é apresentado de seguida na Figura 36.

```

public class SubjectProfile : Profile
{
    public SubjectProfile()
    {
        CreateMap<Subject, SubjectDto>().ReverseMap();
    }
}

```

Figura 36 – Classe *SubjectProfile*

Finalmente é criado o *controller*, que recebe e processa os pedidos relativos à gestão de áreas, mais uma vez apenas é apresentado uma porção do *controller* através da Figura 37, aquela responsável pela consulta através do id.

```

[HttpGet("{id}")]
public async Task<IActionResult> GetSubject(long id)
{
    Logger?.LogDebug("'{0}' has been invoked", nameof(GetSubject));

    var response = new SingleResponse<SubjectDto>();

    try
    {
        response.Model = await _subjectService.GetSubjectByID(id);
    }
    catch (Exception ex)
    {
        response.DidError = true;
        response.ErrorMessage = "There was an internal error";

        Logger?.LogCritical("There was an error on '{0}' invocation: {1}", nameof(GetSubject), ex);
    }

    return response.ToHttpResponse(true);
}

```

Figura 37 – Método da classe *SubjectController*

Em conjunto com a implementação de novas entidades e conceitos, também foram acrescentados novos atributos a entidades existentes como foi indicado no domínio do problema. Como por exemplo a classe *Project*, teve de ser atualizada com atributos que representem a data de início e fim do mesmo, em conjunto com um atributo que indique se este se encontra concluído.

#### 4.2.2 Testes

Quanto aos testes realizados sobre este componente mantiveram-se aqueles realizados na versão inicial do GesProj, com a alteração de alguns dos existentes e a criação de novos testes com base nos novos conceitos e funcionalidades.

Ou seja, o componente foi testado com base em testes unitários e de integração. Sendo testes unitários responsáveis por testar uma unidade, método ou classe, isolada do sistema. Enquanto que os testes de integração verificam o correto funcionamento de diferentes unidades em conjunto.

No desenvolvimento dos testes unitários foi utilizada a ferramenta xUnit, que auxilia na construção e execução de testes unitários em *.NET Core*. Na Figura 38, é exemplificado um dos testes implementados, que neste caso verifica se o DTO devolvido pelo método *GetSubjectByID*, da classe *SubjectService*, devolve a informação esperada.

```

[Fact]
public async Task GetSubjectByID_StateUnderTest_ExpectedBehavior()
{
    var Name = "Gestão Técnica";

    /// Arrange
    var dbContext = await DbContextMocker.GetPortalRHDbContextAsync();
    var unitUnderTest = this.CreateService(dbContext);
    long subjectId = 1;

    // Act
    var subResult = await unitUnderTest.GetSubjectByID(
        subjectId);

    dbContext.Dispose();

    // Assert
    Assert.Equal(Name, subResult.Name);
}

```

*Figura 38 – Teste sobre a consulta de Área por id*

No caso dos testes de integração, foi utilizado o Postman que é uma ferramenta que permite testar o componente através de pedidos HTTP. Para os vários pedidos criados para testar o componente, foram criadas as verificações presentes na Figura 39.

```

tests["Has Content-Type"] = responseHeaders.hasOwnProperty("Content-Type");

pm.test("Response has json body", function() {
    pm.response.to.be.withBody;
    pm.response.to.be.json;
});

pm.test("Response time is below 2000ms", function () {
    pm.expect(pm.response.responseTime).to.be.below(2000);
});

// The response body must include an "id" property
var data = JSON.parse(responseBody);
tests["Response has an ID"] = data.id !== null;

```

*Figura 39 – Testes de integração do back-end*

Tendo especificado os testes na Figura 39, estes depois são aplicados quando é realizado um pedido HTTP ao componente abordado neste capítulo, que em caso de sucesso os resultados são exemplificados na Figura 40.

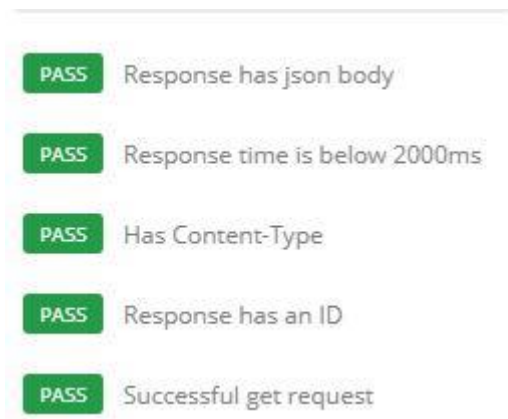


Figura 40 – Resultado dos testes de integração do back-end

### 4.3 PortalRHGecadQuerier

O componente em questão faz parte dos componentes a serem desenvolvidos de raiz para a nova solução. O componente PortalRHGecadQuerier foi implementado com recurso à *framework ASP.NET Core* e mantém uma estrutura semelhante à ilustrada na Figura 31. A estrutura deste componente é apresentada na Figura 41.

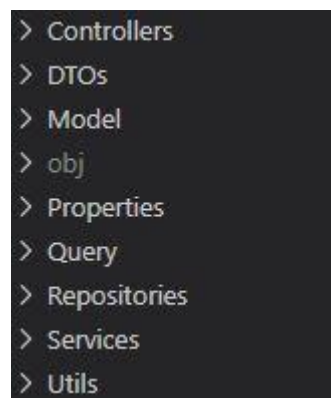


Figura 41 - Estrutura de diretórios do PortalRHGecadQuerier

O diretório *Query* é o único aspeto novo nesta arquitetura, pois é responsável por conter a implementação das diferentes *queries* utilizadas para o preenchimento de *dashboards*.

Tal como no componente anterior e segundo os Requisitos Não Funcionais, foram implementados *logs*, utilizando a mesma lógica que a da Figura 37. Onde no *controller* foram colocadas as instruções necessárias para em caso de um erro interno apenas é enviada uma mensagem de erro genérica e a mensagem detalhada insere-se num registo de *logs*.

Para melhor explicar a implementação das diferentes funcionalidades, este capítulo é dividido em diferentes secções.

### **4.3.1 Implementação das Consultas**

Começa-se por abordar a implementação das diferentes funcionalidades relacionadas com a consulta de informação

#### **4.3.1.1 Consulta de percentagens de afetação dos investigadores aos projetos**

Como foi mencionado na secção de Requisitos Funcionais, deve ser possível consultar as afetações dos investigadores e se necessário alterar a granularidade desta consulta. Para possibilitar esta granularidade foram implementados diferentes métodos, que permitem:

1. Consultar as afetações de todos os investigadores a todos os projetos, dentro de um período de tempo.
2. Consultar as afetações de todos os investigadores a um projeto, dentro de um período de tempo com base no id do projeto.
3. Consultar as afetações de um investigador a todos os projetos, dentro de um período de tempo com base no id do investigador.
4. Consultar as afetações de um investigador a um projeto, dentro de um período de tempo com base no id do investigador e do projeto.

Como estes métodos são acedidos através do *controller* com recurso a pedidos HTTP GET, tanto os ids e datas que delimitam o período são recebidos através do *url* do pedido, como por exemplo "*api/Query/Investigator/1/Project/2/01-06-2019/30-06-2022*". Este *controller* acede aos métodos disponibilizados pelo serviço que acede aos métodos contidos na classe *AffectationQuery* do diretório *Query*.

Para evidenciar a implementação deste caso de uso, serão de seguida abordados os vários métodos responsáveis por permitir a consulta da afetação de todos os investigadores a todos os projetos. Começando pelos métodos presentes no *controller* e serviço através da Figura 42 e Figura 43 respetivamente. Na Figura 42 também são apresentados os comandos relativos ao tratamento de *logs*.

```
[HttpGet("Investigator/Project/{inDate}/{endDate}")]
public async Task<IActionResult> GetInvestigatorsProjectsAffectation(string inDate, string endDate)
{
    Logger?.LogDebug("'{}' has been invoked", nameof(GetInvestigatorsProjectsAffectation));

    var response = new ListResponse<InvestigatorAffectationGlobalDto>();

    try
    {
        response.Model = await _queryService.GetInvestigatorsProjectsAffectation(inDate, endDate);
    }
    catch (Exception ex)
    {
        response.DidError = true;
        response.ErrorMessage = "There was an internal error";

        Logger?.LogCritical("There was an error on '{}{}' invocation: {1}", nameof(GetInvestigatorsProjectsAffectation), ex);
    }

    return response.ToHttpResponse();
}
```

Figura 42 – Método da classe QueryController

```
public async Task<ICollection<InvestigatorAffectationGlobalDto>> GetInvestigatorsProjectsAffectation(string inDate, string endDate)
{
    return await affectationQuery.GetInvestigatorsProjectsAffectation(inDate, endDate);
}
```

Figura 43 – Método de Afetações da classe QueryService

Como é evidente pelo código apresentado e pelo que foi mencionada anteriormente a lógica da consulta em si encontra-se na classe *AffectationQuery*, como é apresentado de seguida na Figura 44.

```
public async Task<ICollection<InvestigatorAffectationGlobalDto>> GetInvestigatorsProjectsAffectation(string inDate, string endDate)
{
    var listDTO = new List<InvestigatorAffectationGlobalDto>();
    var listInvestigator = await investigatorRepository.GetAllInvestigatorsAsync();

    foreach (var investigator in listInvestigator)
    {
        listDTO.Add(await GetInvestigatorProjectsAffectation(investigator, inDate, endDate));
    }

    return listDTO;
}
```

Figura 44 – Método da classe AffectationQuery

Através do método *GetInvestigatorProjectsAffectation*, são consultadas as afetações do investigador a cada projeto no período indicado. Após se ter adquirido as afetações de todos os projetos é calculada a afetação global no período, ou seja, a soma das afetações de todos os projetos. Resultando assim num conjunto de informação representado pelo *InvestigatorAffectationGlobalDto*, que para lá do nome e id do investigador também contém as afetações do investigador em cada projeto existente no sistema e a afetação global do investigador no período especificado.

#### 4.3.1.2 Consulta da *timeline* de um projeto

Tal como é apresentado na secção de Requisitos Funcionais, deve ser possível consultar a *timeline* de um projeto, incluindo o ponto de situação do mesmo.

Segundo o que foi especificado na Figura 29, a consulta da *timeline* de um projeto é realizada através do id do mesmo, o que se traduz num método no *controller* do componente que é acedido através de um *url* com o formato “*api/Query/Project/Timeline/2*”, onde 2 é o id. O *controller* faz o pedido ao serviço onde é disponibilizado o método da Figura 45.

```
public async Task<ProjectTimelineDto> GetProjectTimelineById(long id)
{
    return await timelineQuery.GetProjectTimelineById(id);
}
```

Figura 45 – Método de *Timeline* da classe *QueryService*

A lógica da consulta está contida na classe *TimelineQuery* do diretório *Query*, e é acedida através do método *GetProjectTimelineById* que devolve uma instância de um *ProjectTimelineDto*. O *ProjectTimelineDto* representa um conjunto de informação que caracteriza o período de tempo abrangido pelo projeto, a lista das suas atividades e a lista dos seus eventos.

O conceito de evento, foi criado durante o processo de desenvolvimento e tornou-se numa necessidade para possibilitar uma melhor integração com o *front-end*, mas isso será mais bem explicado numa secção futura. Um evento, neste contexto, indica uma data limite para a realização de *milestones* ou indicadores, ou indica a data de realização de despesas ou indicadores. Como o retorno da consulta é um DTO, um evento é representado pela classe *EventDto*.

Sendo assim no método *GetProjectTimelineById* são consultadas as *milestones*, despesas, indicadores a realizar e realizados do projeto, para depois serem organizados e agrupados pelas diferentes datas e finalmente mantidos no *EventDto*. Também é necessário indicar que o DTO do indicador por realizar, não só contem o valor total a ser realizado, mas também as quantidades já realizadas e por realizar do indicador em questão.

#### 4.3.1.3 Consulta de indicadores por projeto

Como foi mencionado na secção de Requisitos Funcionais, deve ser possível consultar os indicadores realizados e por realizar de um projeto. Sendo assim este caso de uso é muito semelhante no seu funcionamento à consulta da *timeline*.

Primeiro, como foi estabelecido na Figura 30, esta consulta é realizada apenas com base no id do projeto, através de um *url* como “*api/Query/Project/Indicator/2*”, onde o 2 é o id do projeto. Este pedido é processado pelo *controller*, que depois acede ao serviço que por sua parte acede ao método *GetProjectIndicatorById* da classe *IndicatorQuery* do diretório *Query*.

No método *GetProjectIndicatorById* são consultados os indicadores realizados e por realizar do projeto que depois são devolvidos sobre a forma de um *ProjectIndicatorDto*. Este DTO contem o nome e id do projeto em questão, em conjunto com uma lista dos DTOS representantes dos indicadores por realizar do projeto. Estes DTO contem a informação do respetivo indicador por realizar e uma lista dos DTO dos respetivos indicadores realizados.

#### 4.3.2 DTOFactory

A classe *DTOFactory*, substitui os perfis e mapeamento mencionados na secção PortalRHGecadBackEnd. Esta decisão foi tomada devido ao facto de neste componente raramente ocorrer uma transição transparente de entidade para DTO, ou o DTO necessita de informação de origem variada para ser criado.

É verdade que existe a abordagem alternativa de criar classes auxiliares e assim permitir o mapeamento tal como no componente anterior. No entanto no caso do componente em questão não existiu nenhuma situação onde tal abordagem se justificasse, daí se ter optado pela abordagem aqui descrita.

Sendo assim para exemplificar o funcionamento da classe *DTOFactory* considere-se o *YearMonthAffectationDto*, representado na Figura 46.

```

/// <summary>
/// Affectation in a year and month
/// </summary>
public class YearMonthAffectationDto
{
    /// <summary>
    /// Affectation date year
    /// </summary>
    public int Year { get; set; }

    /// <summary>
    /// Affectation month
    /// </summary>
    public int Month { get; set; }

    /// <summary>
    /// Affectation value
    /// </summary>
    public double AffectationValue { get; set; }

    /// <summary>
    /// Investigator name
    /// </summary>
    public string Investigator { get; set; }
}

```

Figura 46 – DTO de Afetação num ano e mês

Para criar este DTO não existe uma entidade correspondente para ser mapeada, logo na classe *DTOFactory* existe um método que recebe a informação necessária para o criar. Este método é apresentado na Figura 47.

```

public static YearMonthAffectationDto YearMonthAffectationDtoDispenser(ExecutiveAffectation executiveAffectation,
    MonthExecutiveAffectation monthExecutiveAffectation, Investigator investigator)
{
    return new YearMonthAffectationDto
    {
        Year = executiveAffectation.Year,
        Month = monthExecutiveAffectation.Month,
        AffectationValue = monthExecutiveAffectation.MonthAffectationValue,
        Investigator = investigator.Name
    };
}

```

Figura 47 – DTOFactory de Afetação num ano e mês

### 4.3.3 Testes

Os testes realizados sobre este componente seguem a mesma mentalidade que os testes desenvolvidos para o PortalRHGecadBackEnd, ou seja, testes unitários com a utilização de xUnit e testes de integração com Postman.

Na Figura 48, é apresentado um excerto do teste unitário realizado sobre o método *GetInvestigatorProjectsAffectationById*, da classe *AffectationQuery*, que permite consultar as afetações de um investigador. Recebendo o id "1", e a data de início e fim do único projeto a ser considerado no contexto é verificado se o DTO resultante devolve informação conforme esperado.

```
var result = await unitUnderTest.GetInvestigatorProjectsAffectationById(1, "16-04-2018", "16-04-2021");
dbContext.Dispose();

var investigator1 = result.ToList()[0];

Assert.Equal(1, investigator1.InvestigatorId);
Assert.Equal("Test Investigator", investigator1.InvestigatorName);
Assert.Single(investigator1.AffectationList);
Assert.Equal(36, investigator1.GlobalAffectationList.Count);
Assert.Equal("Test Project", investigator1.AffectationList[0].ProjectName);
Assert.Equal("Test Activity 1", investigator1.AffectationList[0].ActivityAffectationList[0].ActivityName);
Assert.Equal(36, investigator1.AffectationList[0].ActivityAffectationList[0].AffectationsList.Count);

for (int i = 0; i < 36; i++)
{
    var yearG = investigator1.GlobalAffectationList[i].Year;
    var monthG = investigator1.GlobalAffectationList[i].Month;
    var valueG = investigator1.GlobalAffectationList[i].AffectationValue;

    var yearA = investigator1.AffectationList[0].ActivityAffectationList[0].AffectationsList[i].Year;
    var monthA = investigator1.AffectationList[0].ActivityAffectationList[0].AffectationsList[i].Month;
    var valueA = investigator1.AffectationList[0].ActivityAffectationList[0].AffectationsList[i].AffectationValue;

    Assert.Equal(yearA, yearG);
    Assert.Equal(monthA, monthG);
    Assert.Equal(valueA, valueG);
}
```

*Figura 48 – Porção de um teste unitário sobre a consulta de afetações*

No caso dos testes de integração em Postman, estabeleceu-se uma configuração semelhante à apresentada na Figura 39, mas considerando um maior período de tempo para a resposta do pedido. Tal período deve-se ao maior volume de informação processado quando um pedido é realizado. A configuração destes testes é apresentada na Figura 49 e para cada pedido individual foram acrescentados testes mais específicos.

```

tests["Has Content-Type"] = responseHeaders.hasOwnProperty("Content-Type");

pm.test("Response has json body", function() {
  pm.response.to.be.withBody;
  pm.response.to.be.json;
});

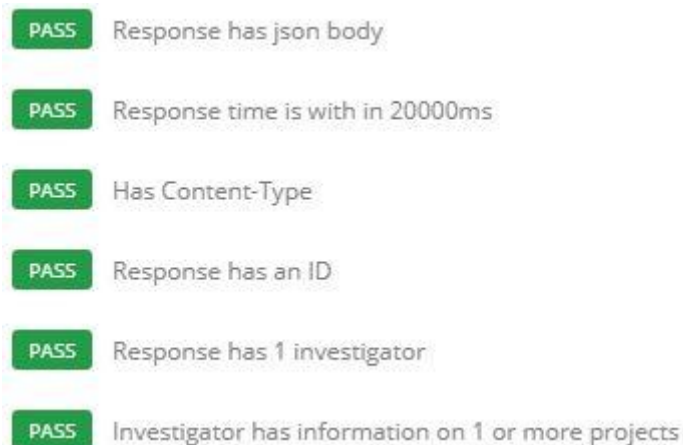
pm.test("Response time is with in 20000ms", function () {
  pm.expect(pm.response.responseTime).to.be.below(20000);
});

// The response body must include an "id" property
var data = JSON.parse(responseBody);
tests["Response has an ID"] = data.id !== null;

```

Figura 49 – Testes de integração do componente PortalRHGecadQuerier

O resultado dos testes realizados sobre o pedido HTTP responsável por consultar a afetação de um investigador pelos vários projetos, encontra-se na Figura 50. Onde em conjunto com os testes da Figura 49, também é verificado se a resposta apenas considera um investigador e vários projetos.



PASS Response has json body  
 PASS Response time is with in 20000ms  
 PASS Has Content-Type  
 PASS Response has an ID  
 PASS Response has 1 investigator  
 PASS Investigator has information on 1 or more projects

Figura 50 – Resultado dos testes de integração do componente PortalRHGecadQuerier

## 4.4 PortalRHGecadRecommender

O componente em questão teve de ser implementado de raiz com recurso à framework *ASP.NET Core*. Com uma estrutura muito semelhante à ilustrada na Figura 31, como é apresentado na Figura 51.

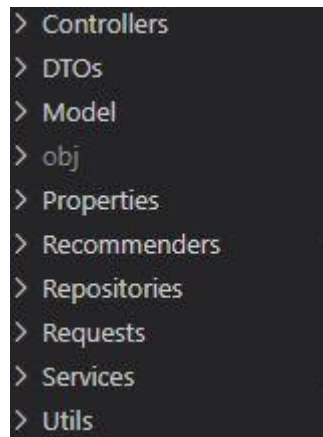


Figura 51 - Estrutura de diretórios do PortalRHGecadRecommender

Nesta estrutura o único diretório cujo conteúdo necessita de ser explicado é o *Recommenders*, que contem a implementação das diferentes abordagens de recomendação e as classes auxiliares utilizadas por estas.

Com o objetivo de criar diferentes abordagens, e no fim do processo de implementação haver uma grande diversidade de abordagens para testar dissidiu-se implementar cinco abordagens, sendo estas:

1. Recomendações com base numa abordagem baseada em conteúdo;
2. Recomendações com base numa abordagem colaborativa (Microsoft.ML);
3. Recomendações com base numa abordagem colaborativa (*k-nearest neighbors algorithm* (KNN));
4. Recomendações com base numa abordagem híbrida (Abordagem 2 + Abordagem 1);
5. Recomendações com base numa abordagem híbrida (Abordagem 3 + Abordagem 1).

Para implementar a abordagem baseada em conteúdo utilizou-se uma implementação focada principalmente na aplicação das regras de negócio e das restrições apresentadas pelo *input* do utilizador.

A primeira implementação com base numa abordagem colaborativa foi feita através da aplicação dos *packages* Microsoft.ML e Microsoft.ML.Recommender, mencionados na secção Tecnologias, pois estes *packages* disponibilizam métodos propícios para a implementação deste tipo de abordagem. Como alternativa à utilização de *packages*, na segunda abordagem colaborativa decidiu-se utilizar uma heurística, mais especificamente KNN.

No caso das abordagens híbridas optou-se por utilizar uma abordagem *cascade* onde os resultados da abordagem colaborativa são melhorados pela abordagem baseada em conteúdo.

Quando um utilizador pretende gerar recomendações faz um pedido HTTP POST que é realizado ao componente e é recebido pelo único método presente no único *controller* existente no componente. Como este método representa um pedido POST, o pedido que este recebe tem um corpo associado, que contém a informação relativa às preferências do utilizador incluindo o método de recomendação a utilizar.

Tal como no componente PortalRHGecadQuerier, no *controller* foram colocadas as instruções necessárias para em caso de um erro interno apenas ser enviada uma mensagem de erro genérica e a mensagem detalhada ser guardada num registo de *logs*.

Após o *controller* validar o conteúdo do pedido, verificando que a data de início é inferior à data de fim do período de recomendação, que o utilizador especificou um número de investigadores maior que 0 e que a percentagem de afetação limite, ou limite de afetação, encontra-se entre 0 e 1, é invocado um método no serviço. O método em questão é apresentado de seguida na Figura 52.

```
public async Task<ICollection<RecommendationDto>> GenerateRecommendations(RecommendationRequest recommendationRequest)
{
    switch (recommendationRequest.Recommender)
    {
        case 1:
            return await GenerateRecommendationsContentBase(recommendationRequest);
        case 2:
            return await GenerateRecommendationsWithPrediction(recommendationRequest);
        case 3:
            return await GenerateRecommendationsWithKNN(recommendationRequest);
        case 4:
            return await GenerateRecommendationsHybridWithPrediction(recommendationRequest);
        case 5:
            return await GenerateRecommendationsHybridWithKNN(recommendationRequest);
        default:
            return new List<RecommendationDto>();
    }
}
```

Figura 52 – Método de geração de recomendações do RecommenderService

Com base no atributo *Recommender* do pedido, é determinado qual a abordagem de recomendação a ser utilizada, que implica a invocação de métodos presentes nas várias classes do diretório *Recommenders*. Para melhor explorar a implementação destas abordagens, o capítulo é dividido em diferentes secções.

#### 4.4.1 Implementação das Abordagens

Começa-se por abordar a implementação das diferentes abordagens de criação de recomendações.

##### 4.4.1.1 Abordagem baseada em conteúdo

Recomendações baseadas em conteúdo, tem como base as preferências do utilizador em questão no dado contexto. Neste caso não existe um utilizador, apenas o pedido recebido por HTTP POST, logo estas são as preferências que se têm de satisfazer.

Sendo assim, quando é escolhida esta abordagem de recomendação é invocado o método *GenerateRecommendations*, da classe *ContentBaseRecommender* do diretório *Recommenders*. A implementação deste método é apresentada na Figura 53.

```
public async Task<ICollection<RecommendationDto>> GenerateRecommendations(RecommendationRequest recommendationRequest)
{
    var recommendationsMap = new Dictionary<Subject, List<List<TagRecommendationInformation>>>();

    var placementPercentage = recommendationRequest.PlacementPercentage;
    var investigatorNumber = recommendationRequest.InvestigatorNumber;
    var inicialDate = recommendationRequest.InicialDate;
    var endDate = recommendationRequest.EndDate;
    var tags = recommendationRequest.Tags;

    var subjectList = await SharedProcesses.GetSubjectsFromTags(tags, subjectRepository);
    var monthYearList = Utils.Utils.CreateMonthYearList(inicialDate, endDate);

    foreach (var subject in subjectList)
    {
        var invList = (await investigatorRepository.GetInvestigatorsAssociatedWithSubjectsForRecommendationAsync(subject.Id)).ToList();

        var infoMap = await SharedProcesses.CreateInformationMap(
            invList,
            monthYearList,
            executiveAffectationRepository,
            investigatorRepository,
            subject
        );

        var listRec = infoMap.Select(entry => entry.Value).ToList();
        listRec.Sort((l1, l2) => Validators.CompareListTagRecommendationInformationForPriority(l1, l2, placementPercentage));

        recommendationsMap.Add(subject, listRec.Take(investigatorNumber).ToList());
    }

    return DTOFactory.RecommendationDtoListDispenser(recommendationsMap);
}
```

Figura 53 – Método de recomendações baseadas em conteúdo

Para cada uma as *tags* recebidas são adquiridos os investigadores que têm a *tag* em questão no seu perfil científico, que serão considerados para recomendação. No método *CreateInformationMap*, é utilizada essa lista e a lista dos vários meses do período indicado, para adquirir a informação relativa as afetações dos investigadores nesse período. Essa informação contempla para cada mês o valor da sua afetação nesse momento, a sua disponibilidade (1 – o valor da afetação) e o seu *rating* da *tag* (as vezes a que lhe foi afeta até esse mês). O retorno

deste método é utilizado para criar uma lista, que contem para cada investigador a lista da informação das afetações de cada mês.

Esta lista que contem a informação global das afetações dos investigadores é de seguida ordenada através do método *CompareListTagRecommendationInformationForPriority*, que dá a cada lista representante de um investigador a prioridade na recomendação. Para determinar a prioridade de cada investigador são utilizadas as médias da sua afetação e *rating* pois se pretende determinar se ele pode ser considerado uma boa recomendação no período indicado.

Logo, a média de afetação é comparada com o limite de afetação indicado e com o valor da afetação máxima do investigador, e se a média for maior do que um destes o investigador tem uma prioridade menor. O objetivo desta abordagem foi maximizar a possibilidade de ser devolvida uma recomendação, mesmo que algumas das restrições não sejam totalmente cumpridas.

Caso ambos os investigadores a serem comparados passem estas validações, de modo a se considerar tanto as afetações e *rating* no processo de decisão para determinar a prioridade são utilizadas as respetivas proporções entre a média dos seus *ratings* e a média das suas afetações. A implementação deste método é apresentada na Figura 54.

```

public static int CompareListTagRecommendationInformationForPriority(List<TagRecommendationInformation> list1, List<TagRecommendationInformation> list2, double placementPercentage)
{
    if (list1.Count < 1 && list2.Count < 1)
    {
        return 0;
    }

    if (list1.Count < 1)
    {
        return 1;
    }

    if (list2.Count < 1)
    {
        return -1;
    }

    var max1 = list1[0].Investigator_MaxAffection;
    var max2 = list2[0].Investigator_MaxAffection;

    var rating1 = list1.Sum(x => x.Rating) / list1.Count;
    var rating2 = list2.Sum(x => x.Rating) / list2.Count;

    var affection1 = list1.Sum(x => x.AffectionValue) / list1.Count;
    var affection2 = list2.Sum(x => x.AffectionValue) / list2.Count;

    if (affection1 == 0 || affection2 == 0)
    {
        return rating1.CompareTo(rating2);
    }

    if (affection1 > placementPercentage || affection1 > max1)
    {
        return 1;
    }

    if (affection2 > placementPercentage || affection2 > max2)
    {
        return -1;
    }

    if (rating1 == 0 || rating2 == 0)
    {
        return affection1.CompareTo(affection2);
    }

    var res1 = rating1 / affection1;
    var res2 = rating2 / affection2;

    return res2.CompareTo(res1);
}

```

Figura 54 – Método de prioridade de investigadores

Com a informação dos vários investigadores, ordenada são guardadas as listas dos investigadores mais relevante, com base no número de investigadores indicados. Este processo é realizado para as várias *tags* e após o ciclo ter sido concluído a informação recolhida é devidamente formatada no método *RecommendationDtoListDispenser* da classe *DTOFactory*. Tal classe será explorada noutra secção.

#### 4.4.1.2 Abordagem colaborativa (Microsoft.ML)

Primeiro é necessário abordar os *packages* Microsoft.ML e Microsoft.ML.Recommender. Estes fazem parte da biblioteca ML.NET da Microsoft, que permite adicionar capacidades de ML a aplicações .Net (Quintanilla 2019).

Ao escolher esta abordagem no serviço é invocado o método *GenerateRecommendationsWithPrediction* da classe *CollaborativeBaseRecommender* do diretório *Recommenders*. Uma porção deste método é apresentada na Figura 55.

```
foreach (var subject in subjectList)
{
    var infoMap = await SharedProcesses.CreateInformationMap(
        invList,
        monthYearList,
        executiveAffectionRepository,
        investigatorRepository,
        subject
    );

    var recInvList = GetInvestigatorsByPredictions(infoMap, invList, subject, placementPercentage);

    var newInfoMapList = (recInvList.GroupBy(x => x).ToDictionary(x => x.Key, x => infoMap.GetValueOrDefault(x.Key))).ToList();

    recommendationsMap.Add(
        subject,
        newInfoMapList.Select(
            pair => pair.Value
        ).Take(investigatorNumber).ToList()
    );
}

return DTOFactory.RecommendationDtoListDispenser(recommendationsMap);
```

Figura 55 – Método de recomendações com ML

O funcionamento deste método é semelhante ao método principal da abordagem anterior (Figura 53), no entanto neste contexto são considerados todos os investigadores que podem ser utilizados para recomendações, não sendo utilizados apenas aqueles que têm a *tag* a ser considerada no momento no seu perfil científico.

A lógica que determina a prioridade dos vários investigadores encontra-se no método *GetInvestigatorsByPredictions*. No método em questão é primeiro formatada a informação relativa aos investigadores de modo a criar uma instância da classe auxiliar *InvestigatorRating* para aqueles cuja média de afetação seja menor que a percentagem de afetação indicada pelo

utilizador. Para cada investigador válido a instância da classe *InvestigatorRating* contem o id do investigador, a *tag* e o valor para essa *tag* (proporção entre a média dos seus *ratings* e a média das suas afetações). A formatação desta informação é apresentada através do excerto de código da Figura 56.

```

var infoList = new List<InvestigatorRating>();

double invRating, invAffectation;
float invProp;
int invIndex = 0;

foreach (var invKey in infoMap)
{
    invIndex = invList.IndexOf(invKey.Key);
    invRating = invKey.Value.Sum(x => x.Rating) / invKey.Value.Count;
    invAffectation = invKey.Value.Sum(x => x.AffectationValue) / invKey.Value.Count;

    if (invAffectation < placementPercentage)
    {
        if (invRating != 0 && invAffectation != 0)
        {
            invProp = (uint)(invRating / invAffectation);
        }
        else
        {
            invProp = 0;
        }

        infoList.Add(
            new InvestigatorRating
            {
                Investigator = invKey.Key.Id,
                Subject = subject.Id,
                Value = invProp
            }
        );
    }
}

```

Figura 56 – Formatação de informação para utilização em ML

Tendo formatado a informação, já é possível utilizar os métodos de ML disponibilizados pela Microsoft.ML e Microsoft.ML.Recommender, como apresentado na Figura 57.

```

var trainData = mlContext.Data.LoadFromEnumerable<InvestigatorRating>(infoList);

var options = new MatrixFactorizationTrainer.Options
{
    MatrixColumnIndexColumnName = "InvestigatorE",
    MatrixRowIndexColumnName = "SubjectE",
    LabelColumnName = "Value",
    NumberOfIterations = 20,
    ApproximationRank = 100
};

var pipeline = mlContext.Transforms.Conversion.MapValueToKey(
    inputColumnName: "Investigator",
    outputColumnName: "InvestigatorE")
    .Append(mlContext.Transforms.Conversion.MapValueToKey(
        inputColumnName: "Subject",
        outputColumnName: "SubjectE")
        .Append(mlContext.Recommendation().Trainers.MatrixFactorization(options)));

ITransformer model = pipeline.Fit(trainData);

var predictionEngine = mlContext.Model.CreatePredictionEngine<InvestigatorRating, InvestigatorRatingPrediction>(model);

```

Figura 57 – Aplicação de ML

O objeto *mlContext* representa uma instância da classe *MLContext*, e é responsável por criar o ambiente onde serão realizadas as diferentes operações relacionadas com o modelo (Warren et al. 2019).

Com a informação dos investigadores na variável *infoList* esta é carregada como um *training set* para o modelo a ser construído. Seguidamente são escolhidas as opções onde se especifica que será utilizada uma abordagem de factorização de matrizes para a criação de recomendações. Finalmente, é criado o “*pipeline*” que mapeia os atributos da classe *InvestigatorRating* para criar a matriz, e com a utilização do *training set* é criado o modelo de recomendações (Farragher 2019; Kazmi 2020; Warren et al. 2019).

Com o modelo definido é criado o motor de previsões que já pode ser utilizado para gerar recomendações. Neste caso este foi utilizado para prever a compatibilidade dos investigadores com a *tag* em questão, estas depois são ordenadas por ordem decrescente.

De seguida, como apresentado na Figura 55 são retirados os investigadores mais relevantes, com base no número de investigadores indicado pelo utilizador e a lista de compatibilidades. Este processo é realizado para as várias *tags* e após o ciclo ter sido concluído a informação recolhida é devidamente formatada no método *RecommendationDtoListDispenser* da classe *DTOFactory*.

#### 4.4.1.3 Abordagem colaborativa (KNN)

Esta abordagem é acedida pelo método *GenerateRecommendationsWithKNN*, da classe *CollaborativeBaseRecommender*. Tal como a abordagem anterior são considerados todos os

investigadores que podem ser utilizados em recomendações e o processo é realizado de forma cíclica para cada *tag*.

A lógica desta abordagem está contida no método *GetInvestigatorsByKNN*, que apresenta uma adaptação de uma implementação do KNN para este problema. No contexto de recomendações o KNN é utilizado para determinar a proximidade dos interesses dos vários utilizadores em relação ao utilizador para qual as recomendações são realizadas. Da aí estabeleceu-se que os “interesses” de cada investigador são representados por três valores:

1. Média dos seus ratings.
2. Média das suas afetações.
3. Um indicador da presença da *tag* em questão no seu perfil científico com o valor 0, se não está presente no perfil, ou 1.

Estes valores são fáceis de determinar para os investigadores no sistema, mas têm de ser determinados de outra forma para o utilizador, pois neste caso pretende-se escolher aqueles cujos interesses estão mais próximos do *input*. De modo a criar-se informação capaz de representar os interesses do utilizador foi criado o método presente na Figura 58.

```
private double[] createBaseInformationList(Dictionary<Investigator, List<TagRecommendationInformation>> infoMap, double placementPercentage)
{
    var ratings = infoMap.Values.SelectMany(x => x).Select(x => x.Rating).Distinct();
    double baseRatings = 0;

    if (ratings.Count() > 0)
    {
        baseRatings = ratings.Sum() / ratings.Count();
    }

    return new double[] {
        baseRatings,
        placementPercentage,
        1
    };
}
```

Figura 58 – Método de geração de informação com base em *input*

Como se pode concluir, a “média de afetações” do utilizador é igual ao valor do limite de afetação, o indicador da *tag* é igual a 1 pois é preferível a escolha de um investigador com a *tag* no seu perfil. No caso da “média dos seus ratings” como não pode ser calculada a partir do *input* é igual à média dos *ratings* de todos os investigadores considerados naquele momento, para se aproximar o mais possível de um valor realista.

Com esta informação, e a informação de todos os investigadores o método *GetInvestigatorsByKNN* já pode ser utilizado. Neste método é calculada a proximidade entre a informação representante do utilizador e a informação dos vários investigadores através da utilização da Correlação de Pearson. Também é utilizada a Semelhança do Cosseno caso o

método seja utilizado pelas abordagens híbridas, ou a Correlação de Pearson não consiga determinar a proximidade com um número suficiente de investigadores. As implementações destas métricas baseiam-se em implementações encontradas *online*, que foram adaptadas para serem utilizadas neste componente (Clayton 2018).

As distâncias obtidas, através de um dos métodos de cálculo, são ordenadas por ordem decrescente, sendo depois escolhidos os respetivos investigadores tendo sempre em consideração o número de investigadores limite.

#### 4.4.1.4 Abordagens híbridas

Ambas as abordagens híbridas são semelhantes no seu funcionamento, logo não necessitam de secções individuais, logo de modo a explicar o funcionamento destas aqui será apresentada a abordagem que utiliza KNN.

Esta pode ser acedida pelo método *GenerateRecommendationsWithKNN*, da classe *HybridRecommender*, como é apresentado na Figura 59.

```
public async Task<ICollection<RecommendationDto>> GenerateRecommendationsWithKNN(RecommendationRequest recommendationRequest)
{
    var collaborativeRec = await collaborativeBaseRecommender.GenerateRecommendationsForHybridWithKNN(recommendationRequest);
    var listDTO = contentBaseRecommender.GenerateRecommendationsForHybrid(recommendationRequest, collaborativeRec);
    return listDTO;
}
```

Figura 59 – Método de geração de recomendações híbridas

Segundo o conceito de uma abordagem *cascade*, o KNN retorna a informação referente aos investigadores considerados mais relevantes com base no número de investigadores especificado. Esta informação é depois ordenada pelo método seguinte, de onde são escolhidos os investigadores a serem recomendados.

Em termos da lógica do funcionamento duas abordagens utilizadas para criar a abordagem híbrida estas não sofrem alterações. No entanto na aplicação da abordagem de KNN, com o objetivo de aumentar a diversidade das soluções, se o número de investigadores desejado for menor que 75% do valor do número de investigadores utilizado nas recomendações, o número de investigadores desejado é aumentado por 1.

#### 4.4.1.5 Comparação

Tendo implementado as cinco abordagens é de seguida realizada uma breve comparação destas em termos da sua implementação e comportamento geral. No capítulo seguinte é realizada a

comparação final destas abordagens em termos de como estas se comportam em diferentes cenários e qual pode ser considerada como a melhor.

Primeiro comparamos as abordagens em termos de como estas lidam com novos investigadores no sistema, ou seja, investigadores que não estão afetos a nenhum projeto ou estiveram afetos a projetos no passado. Com as implementações atuais apenas a abordagem baseada em conteúdo garante prioridade para a recomendação destes investigadores, enquanto que as abordagens colaborativas focam-se mais na maximização da alocação de investigadores. Naturalmente as abordagens híbridas, reúnem estes dois comportamentos, mas como aplicam primeiro a lógica colaborativa, a probabilidade de novos investigadores terem prioridade é menor que na abordagem baseada em conteúdo.

Seguidamente considera-se como estas abordagens utilizam o *input* fornecido pelo utilizador, a percentagem de afetação de cada investigador e os seus *ratings*. Começando com a abordagem baseada em conteúdo, como foi mencionado anteriormente esta tem como base a aplicação direta do *input* e tanto a afetação e *rating* dos investigadores são utilizados para determinar quais os investigadores têm prioridade no processo de recomendação.

Enquanto que no caso das abordagens colaborativas são utilizadas abordagens de ML e KNN, que consideram a proporção entre a afetação e o *rating* dos investigadores, e a proximidade das características do investigador ao *input*, respetivamente. Mesmo assim estas abordagens cumprem as limitações apresentadas pelo *input* (por exemplo a limitação da percentagem de afetação), a consideração destas limitações não é tão direta ou restritiva como na abordagem baseada em conteúdo. Mais uma vez as abordagens híbridas são uma união destas considerações, mas o resultado final destas abordagens é determinado pela lógica da abordagem baseada em conteúdo.

#### **4.4.2 DTOFactory**

A classe *DTOFactory*, tem a mesma função que a classe mencionada na secção PortalRHGecadQuerier, no entanto apenas disponibiliza um método.

Esta tem a responsabilidade de agrupar o resultado das recomendações por investigador para depois criar o DTO a ser devolvido ao utilizador. A justificação desta transformação deve-se ao facto de mesmo que a geração de recomendações seja mais simples quando realizada para cada *tag*, é mais simples para o cliente consultar a informação agrupada por investigador.

### 4.4.3 Testes

Os testes realizados sobre este componente seguem a mesma mentalidade que os testes desenvolvidos sobre os componentes anteriores, ou seja, testes unitários com a utilização de xUnit e testes de integração com Postman.

Começando pelos testes unitários, é aqui apresentado o teste realizado sobre o método *GenerateRecommendations* da classe *ContentBaseRecommender*. O objetivo deste teste é determinar se foram recomendados os investigadores esperados e se estes foram associados às *tags* especificadas. O teste em questão é apresentado na Figura 60.

```
[Fact]
0 references | Run Test | Debug Test
public async Task GenerateRecommendations_StateUnderTest_ExpectedBehavior()
{
    var dbContext = await DbContextMocker.GetPortalRHDbContextAsync();
    var unitUnderTest = this.CreateContentBaseRecommender(dbContext);
    var request = this.CreateRecommendationRequest();

    var result = (await unitUnderTest.GenerateRecommendations(request)).ToList();
    dbContext.Dispose();

    Assert.Equal(2, result.Count);

    var investigator1 = result[0];
    var investigator2 = result[1];

    Assert.Equal(1, investigator1.InvestigatorId);
    Assert.Equal("Test Investigator", investigator1.Investigator);
    Assert.Equal(2, investigator1.Recommendations.Count);
    Assert.Equal("Gestão Técnica", investigator1.Recommendations[0][0].Tag);
    Assert.Equal("Gestão Financeira", investigator1.Recommendations[1][0].Tag);

    Assert.Equal(2, investigator2.InvestigatorId);
    Assert.Equal("Test Investigator2", investigator2.Investigator);
    Assert.Equal(2, investigator2.Recommendations.Count);
    Assert.Equal("Gestão Técnica", investigator2.Recommendations[0][0].Tag);
    Assert.Equal("Gestão Financeira", investigator2.Recommendations[1][0].Tag);
}
```

Figura 60 – Teste sobre recomendações

No caso dos testes em Postman foi utilizada a configuração da Figura 49 em conjunto com a verificação do número de investigadores retornado. Também foi especificado o corpo do pedido HTTP POST (Figura 61).

```

{
  "InvestigatorNumber": 3,
  "Tags": [1,2,3,5,7],
  "PlacementPercentage": 0.5,
  "InicialDate": "2019.07.01",
  "EndDate": "2022.06.01",
  "Recommender": 1
}

```

Figura 61 – Corpo de pedido HTTP POST

Tendo realizado o pedido e recebido a respetiva resposta, verificou-se que o teste foi realizado sem erros, como é evidenciado na Figura 62.

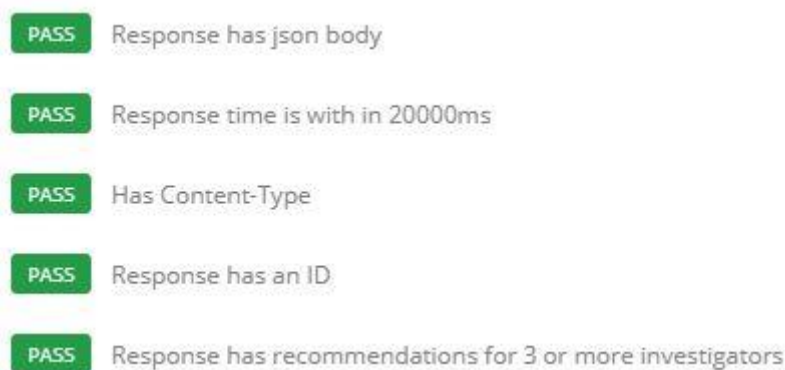


Figura 62 – Resultado dos testes Postman do componente de recomendações

## 4.5 PortalRHGecadFrontEnd

No caso do componente em questão, este já existia como parte do GesProj. No entanto tal como os componentes SQLServerDb e PortalRHGecadBackEnd, este necessita de alterações para acomodar as diferenças no domínio e as novas funcionalidades.

### 4.5.1 GesProj Revisto

Primeiro são apresentadas as alterações realizadas sobre o componente com base no que foi apresentado anteriormente na secção GesProj. Quando um utilizador acede ao portal através de um browser, este é recebido pela página presente na Figura 63. Nesta é feita a listagem dos vários projetos do sistema, com a capacidade de filtrá-los por estado e nome. Também criar um novo projeto ou editar a informação de um projeto existente, incluindo passar este para um estado de concluído.

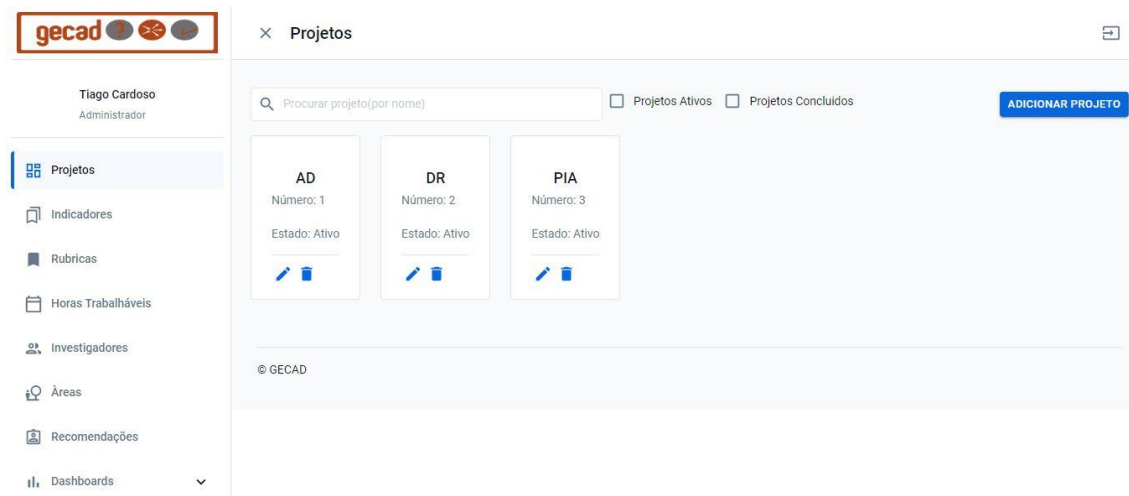


Figura 63 – Nova página inicial do portal

Clicando na opção “Áreas” é apresentada a página, representa na Figura 64. Onde, tal como na página da Figura 9, é realizada a listagem das várias áreas existentes no sistema, com a opção de criar novas áreas ou editar/eliminar áreas existentes.

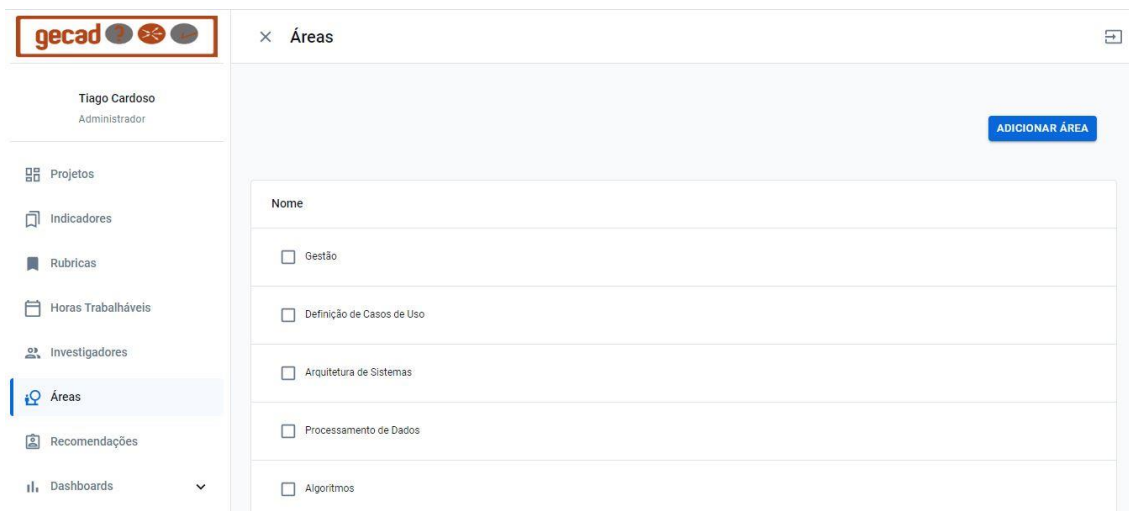


Figura 64 – Página de gestão de áreas

Quanto à página de gestão de um projeto esta mantém-se igual ao que foi apresentado na Figura 5, no entanto foram feitas alterações quando se clica na opção “Indicadores”, ao clicar na opção em questão é apresentada a página aqui representada pela Figura 65, onde o utilizar consegue criar novos indicadores, editar ou eliminar os existentes.

Código	Descrição	Data Limite	Total
222	Relatório	30-06-2022	3
111	Artigo	30-06-2022	1

Figura 65 – Página de gestão de indicadores por realizar

Para cada indicador é dada a opção de o realizar, ao clicar nesta é apresentada uma página com um formato semelhante ao da Figura 65, onde é possível criar indicadores realizados, e editar ou eliminar existentes. Quando à criação de indicadores realizados, esta é feita através da página apresentada na Figura 66, aí é possível associar o indicador a uma *milestone*, indicar o valor realizado, alguma observação e a sua data de realização. Não é possível criar indicadores realizados com uma quantidade superior ao valor por realizar do respetivo indicador por realizar.

Figura 66 – Página de criação de indicadores realizados

Voltando à página de gestão de um projeto, quando se escolhe uma atividade é possível afetar-lhe investigadores ou gerir as suas *milestones*. A página responsável pela gestão de *milestones*

é apresentada na Figura 67. Mais uma vez é possível criar, editar e eliminar *milestones*, no entanto ao editar uma *milestone* também é possível indicá-la como realizada e adicionar uma observação.

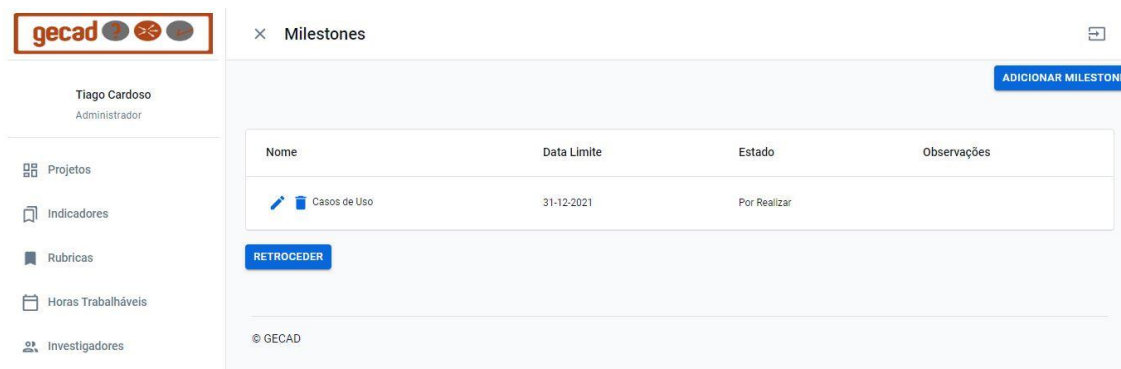


Figura 67 – Página de gestão de milestones

#### 4.5.2 Recomendações

Tendo abordado as alterações relacionadas com o domínio e as novas funcionalidades do PortalRHGecadBackEnd, nesta secção é abordado o aspeto de recomendações no portal. Esta funcionalidade é acedida através da opção na barra lateral denominada “Recomendações”. Quando um utilizador clica nessa opção é lhe apresentada a página presente na Figura 68.

Figura 68 – Página de recomendações de investigadores

O utilizador seguidamente insere as suas preferências quanto às recomendações, como foi especificado na secção Recomendação de recursos humanos dos Processos, e estas têm de ir de encontro às seguintes validações:

- **Investigadores a recomendar:** Tem de ser um número maior que 0;
- **Limite de Afetações dos Investigadores:** Tem de ser um número entre 0 e 1;
- **Início e fim do período de recomendação:** Data de fim tem de ser maior que a data de início;
- **Tags:** Tem de ser selecionada pelo menos uma *tag*.

Só depois do utilizador ter inserido *input* completamente válido é que o utilizador pode gerar recomendações, cujos resultados são apresentados sobre a forma de uma tabela, como é exemplificado na Figura 69.

✕ Recomendação de Investigadores

Investigadores a recomendar  
3

Limite de Afetação dos Investigadores  
0,5

Início do período de recomendação  
20/06/2020

Fim do período de recomendação  
20/07/2020

Gestão Tag

GERAR RECOMENDAÇÕES

Data	Inv6	Inv1	Inv5
2020/6	Tag: Gestão Disponibilidade: 1	Tag: Gestão Disponibilidade: 0.6018	Tag: Gestão Disponibilidade: 0.8403
2020/7	Tag: Gestão Disponibilidade: 1	Tag: Gestão Disponibilidade: 0.6018	Tag: Gestão Disponibilidade: 0.8403

Figura 69 – Página de recomendações com resultados

### 4.5.3 Dashboards

Quanto às diferentes *dashboards* implementadas, estas podem ser acedidas através da opção “Dashboards” da barra lateral, que apresenta uma lista dos diferentes tipos de *dashboards* existentes no sistema, como é apresentado na Figura 70.

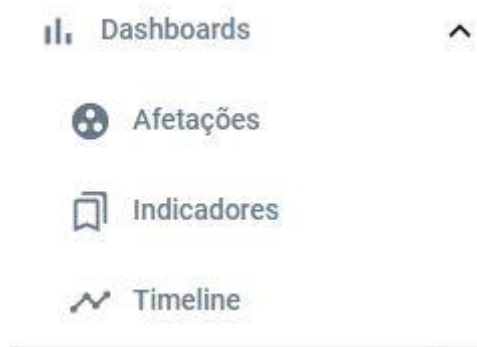


Figura 70 – Componente de seleção de dashboards

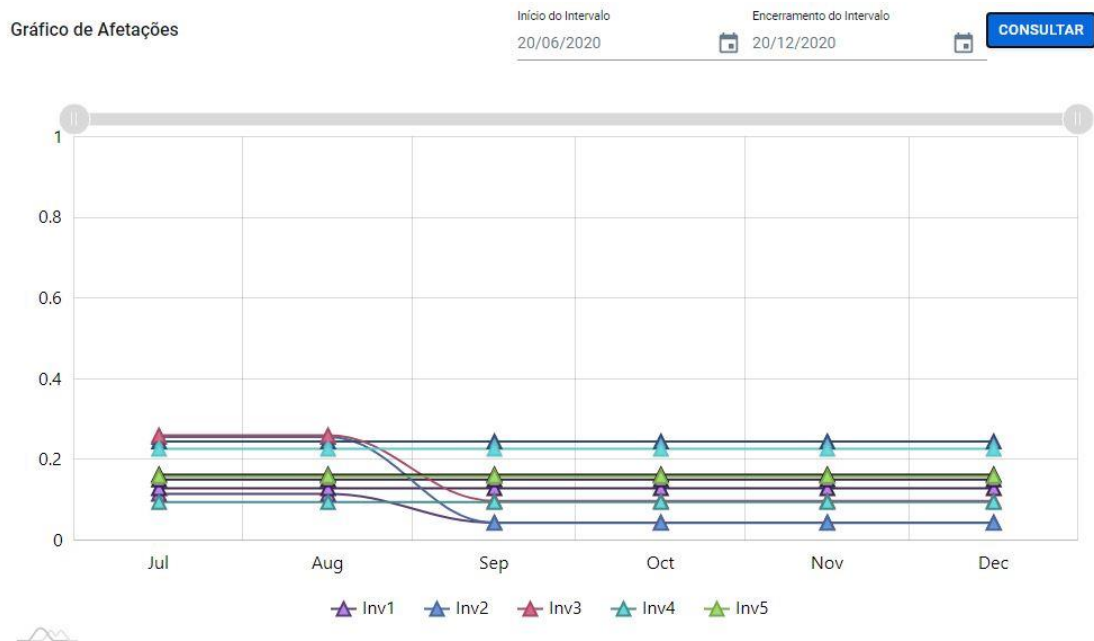
A razão para as *dashboards* serem divididas por tipo é simplicidade, mais especificamente a simplicidade da interpretação de informação por parte do utilizador, tendo em conta a diversidade da informação em questão e a dimensão dos gráficos e tabelas utilizados.

#### 4.5.3.1 Afetações

Para apresentar as afetações dos investigadores foi utilizado um gráfico de linha e uma tabela. Para implementar o gráfico foi escolhida a biblioteca JavaScript amCharts (Majeris 2020) devido à sua versatilidade.

O gráfico de linha, é definido por um eixo x que representa meses e um eixo y representante das afetações dos investigadores, que varia entre 0 e 1. Cada linha neste gráfico representa a afetação de um investigador a um projeto, ou seja, se um investigador estiver afeto a 3 projetos no período indicado serão apresentas 3 linhas.

Quanto ao conteúdo apresentado no gráfico este é definido por uma data de início e fim estabelecida pelo utilizador que depois são utilizadas no pedido HTTP GET ao componente PortalRHGecadQuerier. Com base no pedido é invocado no *controller* o método responsável por permitir consultar as afetações de todos os investigadores a todos os projetos, dentro de um período de tempo. Um exemplo deste gráfico é apresentado na Figura 71.



*Figura 71 – Gráfico de afetações globais*

Para diferenciar os investigadores cada um tem o seu próprio conjunto de cores. Conjunto, pois, num dado período um investigador pode estar afeto a vários projetos, logo cada linha de um investigador é caracterizada por uma diferente tonalidade da mesma cor.

Se o utilizador desejar consultar apenas a informação de alguns dos investigadores apresentados, este pode clicar no nome daqueles que não interessam para a sua consulta, removendo assim a sua informação do gráfico. Para exemplificar, considera-se a Figura 71, e depois “remove-se” todos os investigadores menos aquele denominado como Inv1, como é apresentado na Figura 72.

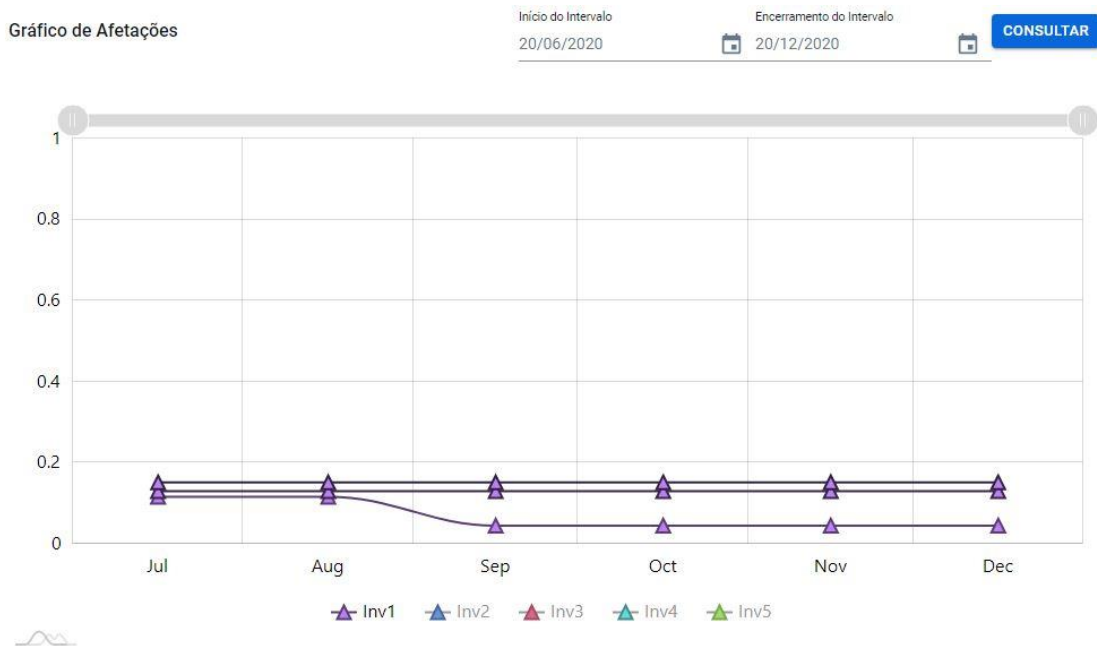


Figura 72 – Gráfico de afetações filtrado

Quanto à tabela de afetações esta apresenta para cada investigador a informação relativa aos projetos que ele está afeto no período indicado e as afetações totais nesse período. Ao contrário do gráfico, esta tabela permite a utilização de todas as consultas referidas na secção de Consulta de percentagens de afetação dos investigadores aos projetos. Para evidenciar esta diversidade na parametrização da tabela, é apresentada a Figura 73, onde para o período indicado são consultadas todas as afetações do investigador Inv1.

Tabela de Afetações		Início do Intervalo	Encerramento do Intervalo		Todos os Projetos		Inv1	CONSULTAR
		20/06/2020	20/12/2020					
Investigador	Projeto	2020/7	2020/8	2020/9	2020/10	2020/11	2020/12	
Inv1	AD	0.1162	0.1162	0.0450	0.0450	0.0450	0.0450	
Inv1	DR	0.1302	0.1302	0.1302	0.1302	0.1302	0.1302	
Inv1	PIA	0.1518	0.1518	0.1518	0.1518	0.1518	0.1518	
Inv1	Total	0.3982	0.3982	0.3270	0.3270	0.3270	0.3270	

Figura 73 – Tabela de afetações

No caso da tabela um investigador não tem um conjunto de cores associado, cada projeto tem uma cor associada. Logo numa listagem para “Todos os Investigadores”, o projeto AD tem sempre a mesma cor na sua respetiva linha.

Quanto à implementação do gráfico e tabela, foi implementada a validação da data do fim do intervalo ser maior que a data de fim, não se podendo realizar uma consulta sem passar esta validação.

#### 4.5.3.2 Indicadores

Para a consulta dos indicadores de projetos, foi implementado um gráfico de barras emparelhadas e uma tabela. Na implementação do gráfico de barras mais uma vez utilizou-se a biblioteca amCharts, onde o eixo x representa cada tipo de indicador do projeto em questão e o eixo y representa a quantidade total do indicador.

Para cada indicador existe uma barra emparelhada, onde parte desta representa a porção por realizar e a outra parte representa o que já foi realizado. Em termos de cor, cada porção da barra é caracterizada por uma tonalidade diferente da mesma cor. Um exemplo deste gráfico é apresentado na Figura 74, para o projeto PIA.

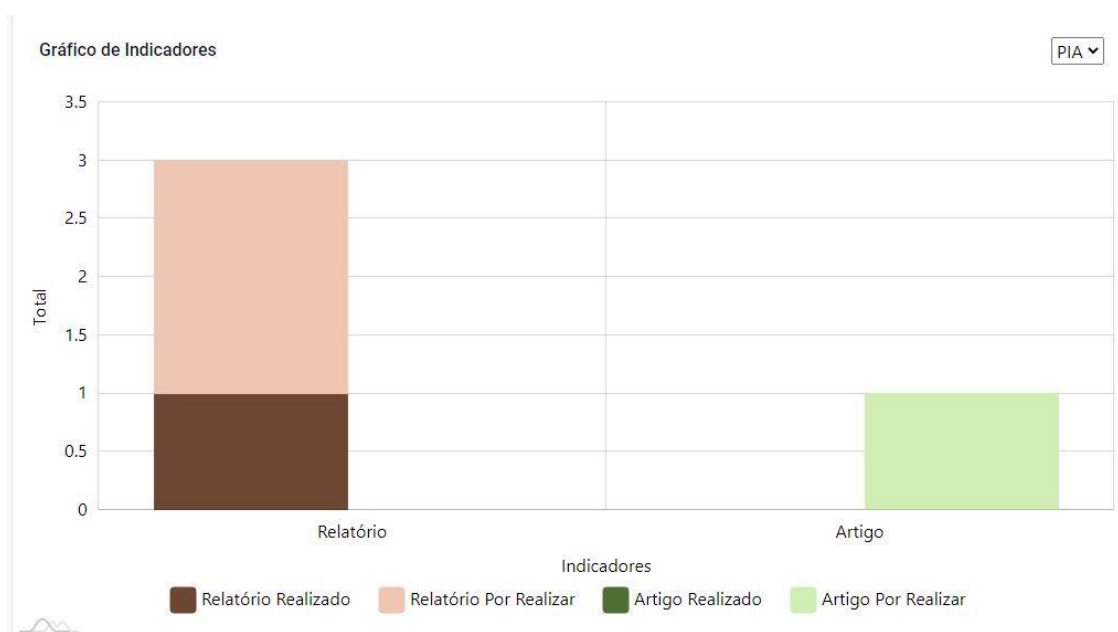


Figura 74 – Gráfico de indicadores

No caso da implementação da tabela, esta foi utilizada como um meio de apresentar para um projeto a informação relativa aos indicadores por realizar e realizados, lado a lado, como é apresentado na Figura 75.

Tabela de Indicadores		
Indicador	Por Realizar	Realizado
Relatório	Quantidade a realizar: 3 Quantidade realizada: 1 Data Limite: 2022/6/30	Quantidade: 1 Data de Realização: 2018/12/20 Milestone:
Artigo	Quantidade a realizar: 1 Quantidade realizada: 0 Data Limite: 2022/6/30	

Figura 75 – Tabela de indicadores

#### 4.5.3.3 Timeline

Para implementar a *timeline* de um projeto, foi utilizado um gráfico denominado “Serpentine Timeline” pela biblioteca amCharts e um exemplo destes é apresentado na Figura 76, para o projeto PIA.

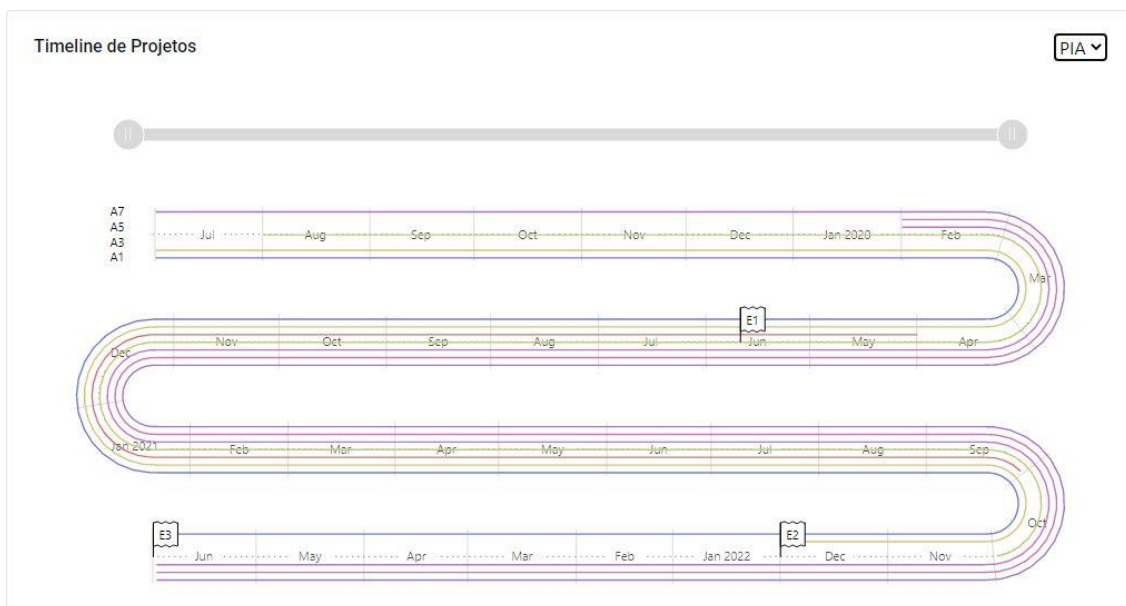


Figura 76 – Gráfico da timeline de um projeto

Cada “linha” do gráfico representa uma das atividades do projeto em questão, e estas em conjunto especificam a duração total do ponto de vista da criação do gráfico. Cada “linha” tem a sua cor, logo é mais fácil para o utilizador as distinguir.

Este gráfico também permite anotar diferentes *flags*, que representam eventos, mas não é possível ter *flags* de eventos diferentes na mesma data. Dai a decisão de se criar o conceito de evento no componente PortalRHGecadQuerier, pois permite agrupar as *milestones*, despesas, indicadores a realizar e realizados do projeto com a mesma data de realização/limite na mesma *flag*.

#### 4.5.4 Introduzir Área

Tendo explorado as alterações feitas ao GesProj, é agora apresentado um exemplo do funcionamento de um caso de uso por de trás das páginas apresentadas. Para esta demonstração escolheu-se o caso de uso responsável por introduzir uma nova área no sistema, por este ser um novo caso de uso e por demonstrar uma interação com um outro componente.

Tal como na solução original, esta nova funcionalidade foi implementada através da utilização do Redux no *React*. O Redux permite a gestão dos vários estados do portal, o que simplifica o processo de codificação. Este é composto por quatro componentes principais:

1. *Reducer*
  - a. Lida com todas as ações.
  - b. Atualiza os estados.
2. *Store*:
  - a. Possui todos os estados.
  - b. Composto pelos *reducers*.
3. *Action*
  - a. Funções que realizam uma ação no software.
  - b. Enviam dados a um *reducer*.
  - c. Para os *reducers* distinguirem a ação a ser cometida, a *action* possui a propriedade *type*.
4. *Dispatch*
  - a. Processa as *actions*.

No caso da criação de uma nova área, segundo a estrutura na Figura 64, o utilizador clica no botão “Adicionar Área” onde lhe é apresentada a *toolbar* com a opção de inserir o nome da

nova área e o botão “Adicionar”. Ao introduzir o nome e clicar no botão é espoletado o *dispatch* da ação para criar a nova área e esta é redirecionada para o respetivo método Figura 77.

```

handleAddSubject = () => {
  const { name } = this.state;

  if (name.length === 0) {
    alert("Campo Nome vazio.");
  } else {
    const { dispatch } = this.props;

    const rubric = {
      name: name
    };

    dispatch(subjectAction.addSubject(rubric));
  }
};

```

Figura 77 – Método invocado pelo botão “Adicionar”

Na ação (subjectAction) é realizado o dispatch de outras ações, presentes na Figura 78, responsáveis por controlar o estado do processo de criação e é realizado o pedido HTTP ao componente PortalRHGecadBackEnd.

```

export const ADD_SUBJECT_SUCESSS = "ADD_SUBJECT_SUCESSS";
export const ADD_SUBJECT_ERROR = "ADD_SUBJECT_ERROR";
export const ADD_SUBJECT_PENDING = "ADD_SUBJECT_PENDING";

function addSubject(data) {
  const payload = new SubjectPostRequest(data.name);
  return dispatch => {
    dispatch(addSubjectPending());
    let apiEndpoint = 'subject';
    basicService.post(apiEndpoint, payload)
      .then((response) => {
        dispatch(addSubjectSucess(response.data.model));
        dispatch(fetchSubjects());
      }).catch((err) => {
        console.log(err);
        dispatch(addSubjectError("Internal Error."));
      })
  }
}

function addSubjectSucess(subject) {
  return {
    type: ADD_SUBJECT_SUCESSS,
    subject: subject
  }
}

function addSubjectPending() {
  return {
    type: ADD_SUBJECT_PENDING
  }
}

function addSubjectError(error) {
  return {
    type: ADD_SUBJECT_ERROR,
    error: error
  }
}

```

Figura 78 – Ações envolvidas na criação de uma nova área



# 5 Avaliação

Neste capítulo é realizada a avaliação da solução desenvolvida. Para tal ser possível primeiro são definidas as hipóteses, metodologias de avaliação e testes estatísticos a serem utilizados no processo de avaliação. Finalmente é realizada a avaliação da solução desenvolvida.

## 5.1 Hipóteses

Com o objetivo de determinar a qualidade e valor da solução foram definidas diferentes hipóteses, que abrangem o sistema de recomendação e *dashboards*:

- O sistema de recomendação é capaz de gerar resultados que cumprem as restrições apresentadas pelo *input* fornecido pelo utilizador.
- O componente de consulta de informação agregada, através de *dashboards*, é capaz de apresentar informação de uma forma precisa e compreensível para o utilizador.

## 5.2 Metodologia de Avaliação

Para o sistema ser avaliado é necessária informação para este utilizar, sendo esta baseada em registos mantidos pelo GECAD relativos à gestão de diferentes projetos e dos investigadores que lhes foram afetados. A informação utilizada é relativa a três projetos e cinco investigadores afetos a estes projetos, havendo assim no sistema a informação necessária para a criação de recomendações e utilização de *dashboards*.

Para avaliar a solução, de um ponto de vista do funcionamento do componente de consulta de informação agregada e satisfação do cliente, foi escolhido um inquérito de satisfação. Esta escolha justifica-se pelo facto de um inquérito ser respondido pelo cliente e assim permite determinar se o cliente está satisfeito com a solução, especialmente com o componente de consulta de informação agregada.

Como respostas às questões presentes no inquérito, foram especificadas as seguintes possibilidades:

- Muito insatisfeito.
- Insatisfeito.

- Sem opinião.
- Satisfeito.
- Muito satisfeito.

As questões do inquérito em si, são apresentadas de seguida:

1. A solução é simples e intuitiva de utilizar?
2. O sistema de recomendação apresenta resultados úteis e diversificados?
3. A escolha de recursos humanos com base em recomendações é simples?
4. As *dashboards* apresentam toda a informação necessária?
5. As *dashboards* apresentam informação de forma completa e consistente?
6. As *dashboards* apresentam informação de uma forma simples de compreender?
7. A solução vai de encontro às necessidades do cliente?

Em conjunto com o inquérito, serão realizados diferentes testes para verificarem a versatilidade e adaptabilidade do sistema de recomendação desenvolvido.

Estes testes têm como base a geração sistemática de recomendações com base num dado *input* e informação presente na base de dados, para verificar se o sistema consegue apresentar recomendações que sigam as restrições apresentadas pelo *input* e estas apresentem alguma diferenciação entre si. Para cada um dos resultados obtidos são calculadas as Métricas de Precisão do Apoio à Decisão (Precisão, *Recall*, *F-Measure*), com o objetivo de determinar a capacidade do sistema de ajudar o utilizador na sua tomada de decisão.

### 5.3 Testes Estatísticos

Tendo obtido uma grande quantidade e variedade de informação através dos diferentes métodos de avaliação é necessário analisá-la para se conseguir tirar conclusões concretas que permitam comprovar ou desmentir as hipóteses apresentadas.

No caso do inquérito de satisfação, a análise dos dados obtidos será realiza com base no cálculo das médias obtidas e apresentação dos resultados com base em gráficos de barras. Em conjunto com esta análise também será estudada a confiabilidade entre avaliadores, que representa um meio de medição da concordância de diferentes intervenientes relativamente ao valor de um dado tema ou tópico (McHugh 2012).

No caso dos testes aplicados ao sistema de recomendação em si, tendo em conta que se trata de uma amostra pequena com informação emparelhada relativa a mais que duas execuções da solução, escolheu-se a aplicação de um teste de Friedman. Através do teste de Friedman é possível comparar os resultados obtidos pela aplicação das métricas aos vários resultados de modo a verificar a hipótese proposta para o sistema de recomendação (McCrum-Gardner 2008).

## **5.4 Sistema de Recomendação**

Nesta secção é abordada a avaliação do sistema de recomendação desenvolvido como parte da solução. Primeiro é abordada a informação utilizada para a avaliação e depois a avaliação em si.

### **5.4.1 Cenários de Teste**

Para conseguir avaliar o sistema foi utilizada informação real, referente a três projetos atualmente em desenvolvimento pelo GECAD. No entanto de modo a impedir a divulgação de informação sensível, o nome dos projetos e investigadores foi alterado. Começa-se por na Tabela 13 apresentar a informação geral dos projetos considerados.

Tabela 13 – Projetos para Avaliação

<b>Projeto</b>	<b>Data de início</b>	<b>Data de fim</b>	<b>Investigadores</b>	<b>Tags</b>	<b>Número de atividades</b>
AD	01-09-2018	31-08-2021	Inv1, Inv2, Inv3	Algoritmos, Arquitetura de Sistemas, Demonstradores, Disseminação, Gestão, Privacidade, Processamento de Dados	7
DR	01-07-2019	31-12-2021	Inv1, Inv4, Inv5	Definição de Casos de Uso, Disseminação, Gestão, Gestão de Conhecimento	6
PIA	01-07-2019	30-06-2022	Inv1, Inv2, Inv3, Inv4	Algoritmos, Arquitetura de Sistemas, Demonstradores, Disseminação, Gestão, Processamento de Dados	6

Segundo a Tabela 13, existem cinco investigadores no sistema, no entanto por motivos de teste foi criado o investigador Inv6, que não está relacionado com nenhum projeto, de modo a avaliar a capacidade da solução de lidar com novos investigadores. Tendo estabelecido todos os investigadores, os seus perfis são apresentados na Tabela 14.

Tabela 14 – Investigadores para Avaliação

Investigador	Limite de Afetação	Perfil Científico
Inv1	0.8	Algoritmos, Arquitetura de Sistemas, Definição de Casos de Uso, Demonstradores, Disseminação, Gestão, Gestão de Conhecimento, Processamento de Dados
Inv2	0.8	Algoritmos, Arquitetura de Sistemas, Definição de Casos de Uso, Demonstradores, Disseminação, Gestão, Processamento de Dados
Inv3	0.8	Algoritmos, Arquitetura de Sistemas, Definição de Casos de Uso, Demonstradores, Disseminação, Gestão, Processamento de Dados
Inv4	0.8	Algoritmos, Arquitetura de Sistemas, Definição de Casos de Uso, Demonstradores, Disseminação, Gestão, Gestão de Conhecimento, Processamento de Dados
Inv5	0.8	Definição de Casos de Uso, Disseminação, Gestão, Gestão de Conhecimento
Inv6	0.8	Algoritmos, Arquitetura de Sistemas, Definição de Casos de Uso, Demonstradores, Disseminação, Gestão, Gestão de Conhecimento, Processamento de Dados

Como foi indicado na secção PortalRHGecadRecommender, foram implementadas cinco abordagens para a realização de recomendações, logo é necessário comparar a capacidade das cinco de ir de encontro às necessidades do cliente. Para facilitar este processo cada abordagem foi atribuída um nome por qual vão ser referenciadas a partir daqui:

1. **A1:** Abordagem baseada em conteúdo;
2. **A2:** Abordagem colaborativa (Microsoft.ML);
3. **A3:** Abordagem colaborativa (KNN);
4. **A4:** Abordagem híbrida (A2 + A1);

5. **A5:** Abordagem híbrida (A3 + A1).

Para comparar cada uma destas abordagens, decidiu-se criar cinco cenários de teste diferentes de modo a adquirir informação relativa aos resultados de como cada abordagem se comporta em diferentes contextos. Cada cenário é caracterizado por um *input* diferente, como é estabelecido na Tabela 15.

Tabela 15 – Tabela de cenários

Cenário	Número de investigadores	Tags	Percentagem de afetação	Data de início	Data de fim
1	3	Gestão, Definição de Casos de Uso, Arquitetura de Sistemas, Algoritmos, Disseminação	0.16	01-09-2020	31-08-2023
2	3	Gestão, Definição de Casos de Uso, Arquitetura de Sistemas, Algoritmos, Disseminação	1	01-09-2020	31-08-2023
3	4	Gestão, Definição de Casos de Uso, Arquitetura de Sistemas, Algoritmos, Disseminação	0.16	01-07-2019	30-06-2022
4	4	Gestão, Definição de Casos de Uso, Arquitetura de Sistemas, Algoritmos, Disseminação	1	01-07-2019	30-06-2022
5	5	Gestão, Definição de Casos de Uso, Arquitetura de Sistemas, Algoritmos, Disseminação	1	01-09-2023	31-08-2025

Para cada resultado dos cinco cenários são calculadas as Métricas de Precisão do Apoio à Decisão (Precisão, *Recall*, *F-Measure*). Segundo a secção de Métricas de Avaliação para calcular a Precisão e *Recall* é necessário determinar o “Nº de *items* relevantes recomendados”, o “Nº de *items* recomendados” e o “Nº total de *items* relevantes”.

O valor do “Nº de *items* recomendados” é simples de calcular pois é igual ao número de investigadores devolvidos por cada abordagem em cada cenário. Quando ao valor do “Nº de *items* relevantes recomendados” este é influenciado pelos investigadores que se consideram relevantes para cada cenário. Sendo assim o valor do “Nº total de *items* relevantes” varia com cada cenário:

- **Cenário 1:** São considerados relevantes 4 investigadores sendo estes Inv6, Inv5, Inv3, Inv2.
- **Cenário 2:** São considerados relevantes 4 investigadores sendo estes Inv6, Inv5, Inv3, Inv2.
- **Cenário 3:** São considerados relevantes 3 investigadores sendo estes Inv6, Inv5, Inv4.
- **Cenário 4:** São considerados relevantes 3 investigadores sendo estes Inv6, Inv5, Inv4.
- **Cenário 5:** São considerados relevantes todos os 6 investigadores, pois nenhum está associado a um projeto no período indicado.

#### 5.4.2 Análise de Resultados

Tendo estabelecido os diferentes cenários, estes são executados uma vez para cada abordagem obtendo-se os resultados presentes no Anexo II, o que possibilita o cálculo das métricas.

Para exemplificar como estas foram calculadas para todas as abordagens e cenários, são de seguida apresentados os cálculos para as métricas da abordagem A1 do Cenário 1. Primeiro são determinados os elementos a serem utilizados no cálculo da Precisão e *Recall*:

- Nº de *items* recomendados: Neste cenário foram recomendados os investigadores Inv6, Inv1, Inv5, Inv3, Inv2, logo o valor é 5.
- Nº de *items* relevantes recomendados: Os investigadores considerados relevantes são Inv6, Inv5, Inv3, Inv2, logo o valor é 4.
- Nº total de *items* relevantes: Como foi estabelecido anteriormente este valor é 4.

Sendo assim as três métricas podem ser calculadas por:

$$\text{Precisão} = \frac{\text{Nº de } \textit{items} \text{ relevantes recomendados}}{\text{Nº de } \textit{items} \text{ recomondados}} = \frac{4}{5} = 0.8$$

$$Recall = \frac{N^{\circ} \text{ de itens relevantes recomendados}}{N^{\circ} \text{ total de itens relevantes}} = \frac{4}{4} = 1$$

$$F - Measure = \frac{2 \times Precisão \times Recall}{(Precisão + Recall)} = \frac{2 \times 0.8 \times 1}{0.8 + 1} = 0.8889$$

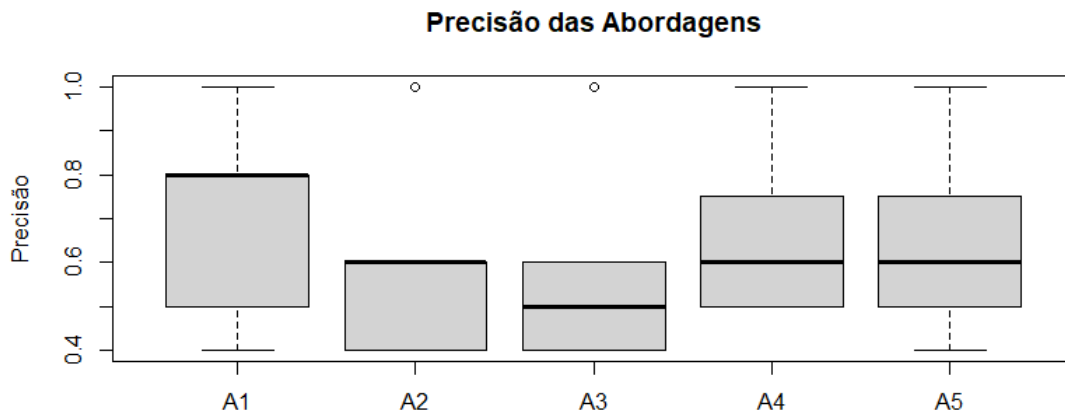
Este processo de cálculo é repetido para todos os cenários e abordagens até se obter a Tabela 16.

Tabela 16 – Tabela de resultados para análise

		Cenário 1	Cenário 2	Cenário 3	Cenário 4	Cenário 5
A1	Precisão	0.8	0.8	0.4	0.5	1
	Recall	1	1	0.6667	1	1
	F-Measure	0.8889	0.8889	0.5	0.6667	1
A2	Precisão	0.6	0.6	0.4	0.4	1
	Recall	0.75	0.75	0.6667	0.6667	1
	F-Measure	0.6667	0.6667	0.5	0.5	1
A3	Precisão	0.6	0.5	0.4	0.4	1
	Recall	0.75	0.5	0.6667	0.6667	1
	F-Measure	0.6667	0.5	0.5	0.5	1
A4	Precisão	0.75	0.6	0.5	0.5	1
	Recall	0.75	0.75	1	1	1
	F-Measure	0.75	0.6667	0.6667	0.6667	1
A5	Precisão	0.75	0.6	0.5	0.4	1
	Recall	0.75	0.75	1	0.6667	1
	F-Measure	0.75	0.6667	0.6667	0.5	1

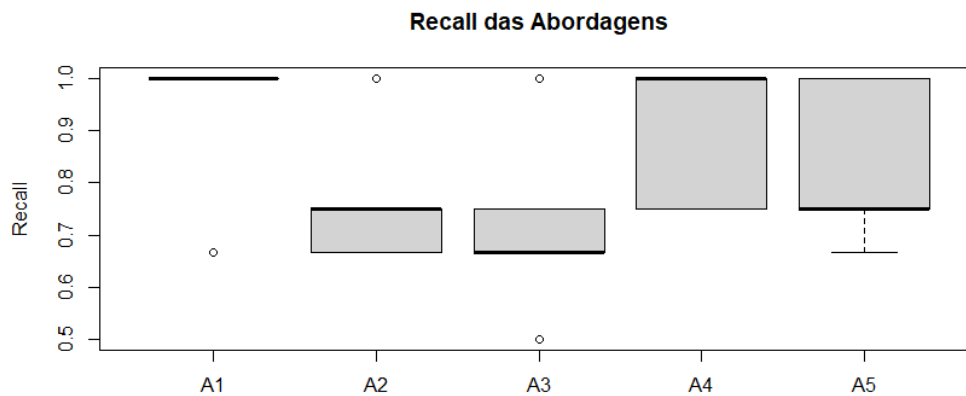
Agora é possível começar o processo de comparação das diferentes abordagens. Em vez de começar com o teste estatístico proposto, são realizadas comparações iniciais com base em

*box-plots*. Como as três métricas utilizadas variam entre 0 e 1, aqui procura-se a(s) abordagem(s) que mais se aproximam de 1.



*Figura 79 – Box-plot das precisões das abordagens*

Observando a Figura 79, é possível determinar que a abordagem A1 apresenta os resultados mais próximos do 1 e tem a maior mediana. No entanto as abordagens A4 e A5 também estão próximas do mesmo conjunto de valores, mesmo que tenham uma mediana menor, logo também podem ser consideradas boas opções em diferentes cenários.



*Figura 80 – Box-plot do recall das abordagens*

Na Figura 80, são apresentados os valores representes do *Recall* das várias abordagens e mais uma vez procura-se a abordagem com os valores mais próximos de 1. Observando o gráfico conclui-se que a abordagem A1 se encontra mais próxima de 1, logo para esta métrica

apresenta os melhores resultados. Também se pode mencionar a abordagem A4, que mesmo tenha uma maior variância de valores, tem a mesma mediana.

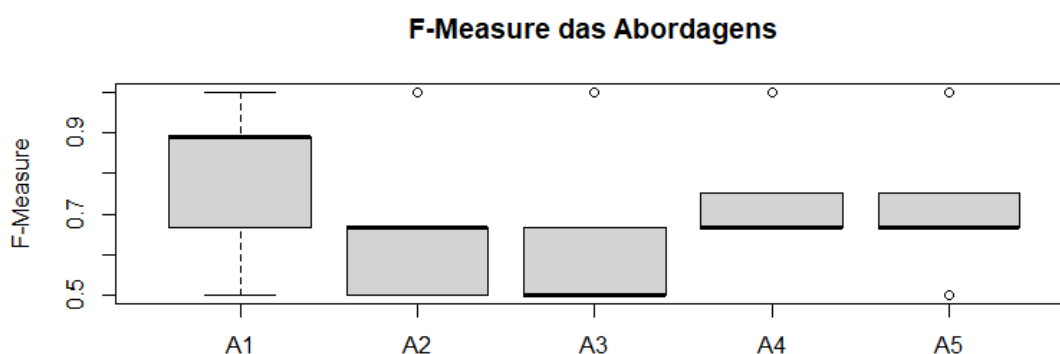


Figura 81 – Box-plot da *f-measure* das abordagens

Observando a Figura 81, tal como nos gráficos anteriores, a abordagem A1 é aquela cujos valores mais se aproximam de 1 e tem a maior mediana, logo esta é considerada a melhor abordagem neste contexto.

Sendo assim, com base na análise dos gráficos apresentados, a melhor abordagem aparenta ser a melhor a A1. No entanto, como proposto, é de seguida realizado um teste para confirmar esta conclusão. Para a realização do teste de Friedman, foi utilizado um *script* em R (Anexo IV) e considerando apenas os valores recolhidos para o *F-Measure* visto que esta reúne as outras duas métricas. Primeiro foi criado um *data frame*, com a mesma estrutura que a Tabela 17.

Tabela 17 – Tabela de *F-Measure* para o teste de Friedman

Cenário	A1	A2	A3	A4	A5
1	0.8889	0.6667	0.6667	0.75	0.75
2	0.8889	0.6667	0.5	0.6667	0.6667
3	0.5	0.5	0.5	0.6667	0.6667
4	0.6667	0.5	0.5	0.6667	0.5
5	1	1	1	1	1

Após a criação do *data frame*, este foi utilizado para criar outro que compacta a informação para criar um *data frame* com as colunas cenário, abordagem e o valor associado, podendo assim já ser utilizado no teste. Para a sua aplicação foi estabelecido um nível de confiança de 95%, com uma alfa de 0.05 e foram criadas as seguintes hipóteses:

- $H_0$ : As abordagens consideradas têm o mesmo desempenho.
- $H_1$ : As abordagens consideradas não têm o mesmo desempenho.

Aplicando a função “*friedman\_test*”, é obtido um *p-value* de 0.0497, ou seja, existem evidências estatísticas que permitem afirmar, ao nível de 95% de confiança, que existem diferenças entre as abordagens. Sendo assim, com a aplicação deste teste e a análise dos *box-plots* anteriores é possível assumir-se que a abordagem A1 tem o melhor desempenho, visto de um ponto de vista geral tem a maior mediana nas três métricas e os valores mais próximos de 1.

Sendo assim é necessário comprovar que de facto A1 pode ser considerada a melhor abordagem, para tal foi escolhido o *Wilcoxon signed rank test* (McCrum-Gardner 2008), para comparar A1 e cada uma das outras abordagens. Para tal manteve-se o nível de confiança de 95%, com uma alfa de 0.05 e estabeleceram-se as seguintes hipóteses:

- $H_0$ : As abordagens consideradas têm o mesmo desempenho.
- $H_1$ : O desempenho da abordagem A1 é maior que o desempenho da abordagem x.

Onde x, representa a abordagem A2, A3, A4 ou A5. Mais uma vez foi utilizado um *script* em R (Anexo IV) para realizar os cálculos necessários, tendo-se obtido a Tabela 18.

Tabela 18 – Tabela de comparação com base em  $p$ -value

Comparação	$p$ -value	Conclusão
A1 e A2	0.0868	$p$ -value > 0.05, logo não há evidencias estatísticas para rejeitar $H_0$
A1 e A3	0.0907	$p$ -value > 0.05, logo não há evidencias estatísticas para rejeitar $H_0$
A1 e A4	0.3946	$p$ -value > 0.05, logo não há evidencias estatísticas para rejeitar $H_0$
A1 e A5	0.2307	$p$ -value > 0.05, logo não há evidencias estatísticas para rejeitar $H_0$

### 5.4.3 Conclusões

Enquanto que através do teste de Friedman foi possível concluir que de facto existe uma diferença entre o desempenho de cada abordagem, quando foi utilizado o *Wilcoxon signed rank test* não foi possível concluir com evidencias estatísticas que a abordagem A1 pode ser considerada a melhor.

Sendo assim, é colocado o foco nos *box-plots* desenvolvidos para comparar as abordagens. Observando os gráficos é possível verificar que as abordagens A2 e A3 no geral têm os menores resultados, logo serão ignoradas. Também se conclui que no Cenário 5, que tem como objetivo comprovar o seu desempenho num período onde ninguém está afeto a um projeto, todos apresentam bom resultados, logo qualquer uma pode ser utilizada.

Assim, apenas se tem de considerar as abordagens A1, A4 e A5. No caso do Cenário 1 e 2 que verificam a capacidade de recomendação num período onde os investigadores, na sua maioria, apenas estão parcialmente afetos, A1 apresenta os melhores resultados. Quanto ao Cenário 3

e 4, que contemplam um período onde todos menos o investigador Inv6 estão afetos, a abordagem A4 apresenta os melhores resultados.

Sendo assim as melhores abordagens podem ser indicadas como a A1 e A4, para determinar qual destas é preferível é utilizada a média do *F-Measure* de cada abordagem que é 0.7889 e 0.75 respetivamente. Logo, como se pretende utilizar a abordagem com os valores mais próximos de 1, a melhor abordagem é a A1.

## 5.5 Dashboards e a Solução em Geral

Nesta secção é abordada a avaliação do componente de consulta de informação agregada e da solução em geral através do inquérito definido anteriormente na secção Metodologia de Avaliação.

Para obter respostas ao inquérito, foi primeiro realizada uma demonstração da solução desenvolvida a 8 investigadores do GECAD e no fim desta foi-lhes pedido que preenchessem uma versão do inquérito presente no Google *Forms*.

### 5.5.1 Inquérito de Satisfação

Enquanto que as respostas ao inquérito se encontram no Anexo III, estas aqui são agrupadas por questão sobre a forma de um gráfico de barras. Começando pela questão de “A solução é simples e intuitiva de utilizar?” na Figura 82.

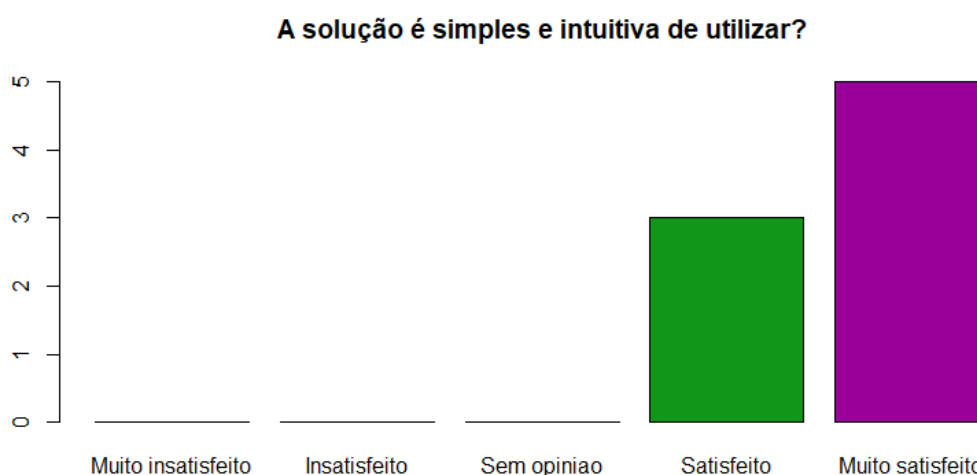
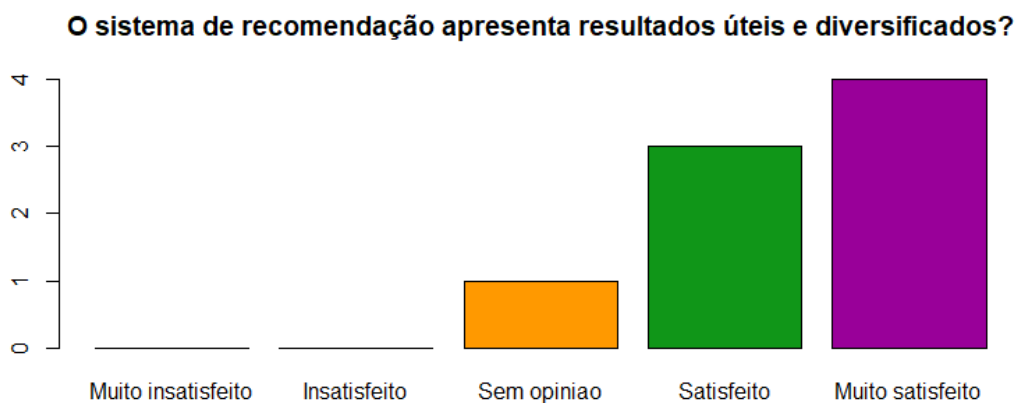


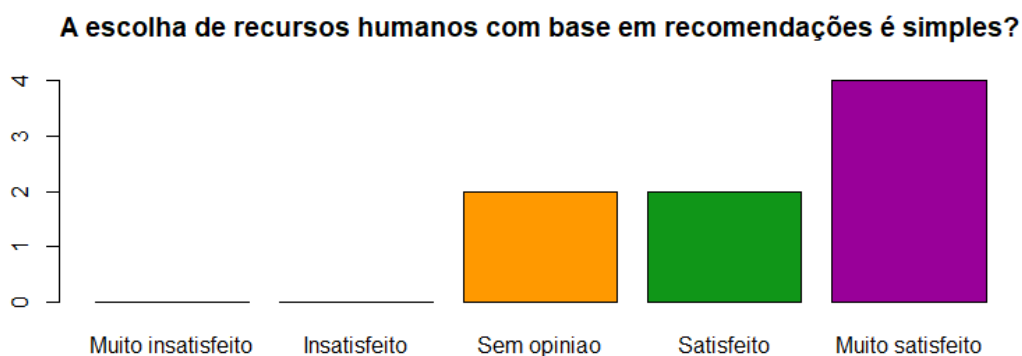
Figura 82 – Gráfico da questão 1

Como é evidenciado pelo gráfico dos 8 investigadores, 3 destes estão satisfeitos e 5 estão muito satisfeitos quanto ao facto da solução ser simples e intuitiva de utilizar. O gráfico da Figura 83 é relativo à questão de se o sistema de recomendação apresenta resultados úteis e diversificados.



*Figura 83 – Gráfico da questão 2*

No caso desta questão, enquanto que um investigador respondeu com “Sem opinião”, 3 indicaram que estavam satisfeitos, enquanto que os restantes 4 indicaram estar muito satisfeitos com este aspeto do sistema de recomendação. As respostas dos investigadores relativamente à capacidade das recomendações ajudarem na escolha de recursos humanos para equipas, são apresentadas na Figura 84.



*Figura 84 – Gráfico da questão 3*

Aqui 4 investigadores indicaram que estão muito satisfeitos, enquanto que 2 indicaram não ter opinião concreta relativamente a esta questão e outros 2 estão satisfeitos.



Figura 85 – Gráfico da questão 4

Passando às questões relativas com as *dashboards*, na Figura 85 são apresentadas a respostas à questão “As *dashboards* apresentam toda a informação necessária?”. Neste caso mais uma vez 4 investigadores estão muito satisfeitos quanto à informação apresentada pelas *dashboards*, 2 estão satisfeitos e os restantes não têm uma opinião concreta.

Quanto à questão “As *dashboards* apresentam informação de forma completa e consistente?”, as respostas são apresentadas na Figura 86.

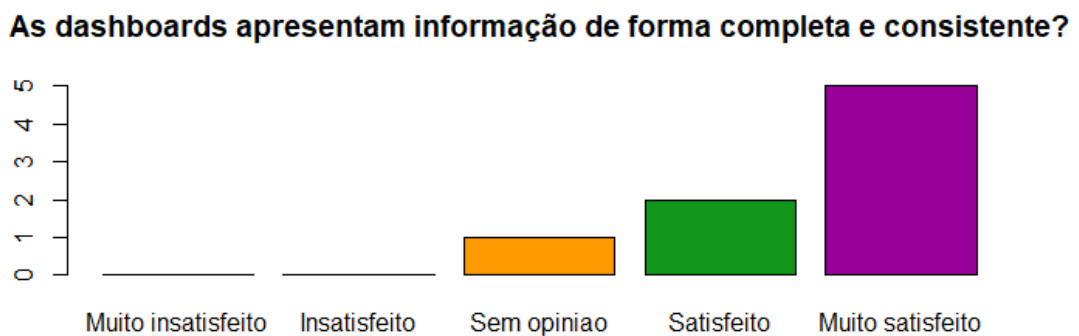
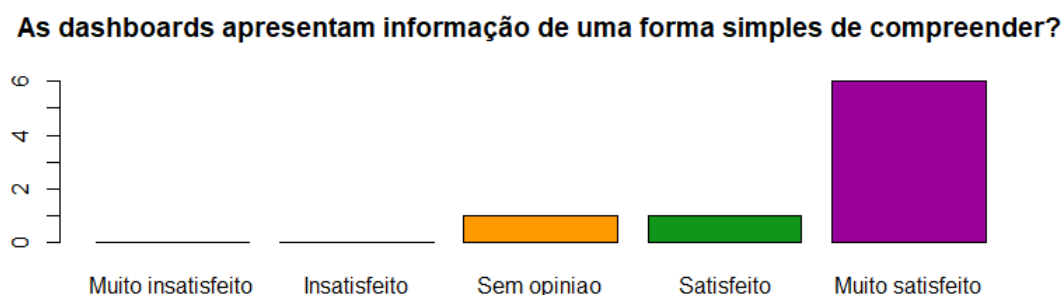


Figura 86 – Gráfico da questão 5

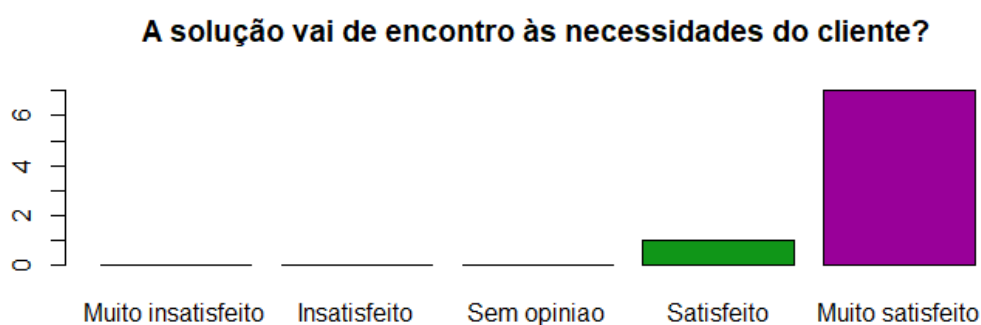
Neste caso apenas um investigador indica não ter opinião, 2 destes estão satisfeitos e mais uma vez a maioria indica estar muito satisfeita. Passando para a questão seguinte, relativa à capacidade das *dashboards* serem capazes de apresentar informação de uma forma compreensível, as respostas são apresentadas na Figura 87.



*Figura 87 – Gráfico da questão 6*

Segundo a Figura 87, 6 investigadores indicaram estar muito satisfeitos com a simplicidade de como a informação é apresentada, enquanto que os 2 investigadores restantes indicaram estar satisfeitos e sem opinião.

Na última questão apresenta no inquérito, é perguntado se a solução como um todo se aproxima das necessidades do cliente.



*Figura 88 – Gráfico da questão 7*

Através da Figura 88, é possível verificar que a maioria dos investigadores, mais especificamente 7 destes, dizem estar muito satisfeitos com a capacidade da solução de ir de encontro às suas necessidades. Com apenas 1 indicando estar apenas satisfeito.

### 5.5.2 Análise de Resultados

Como foi mencionada anteriormente, em conjunto com a análise das respostas apresentadas, também será analisada a confiabilidade entre avaliadores. A confiabilidade também pode ser

referida como a estatística *kappa*, um valor que varia entre -1 e 1, e é avaliada sobre diferentes magnitudes segundo a Tabela 19 (McHugh 2012).

*Tabela 19 – Tabela de kappa (McHugh 2012)*

Valor do <i>kappa</i>	Nível de concordância
0 – 0.20	Nenhum
0.21 – 0.39	Mínimo
0.40 – 0.59	Fraco
0.60 – 0.79	Moderado
0.80 – 0.90	Forte
> 0.90	Quase Perfeito

No contexto desta avaliação são considerados mais do que dois avaliadores, logo será utilizado o *kappa* de Fleiss pois este é utilizado para 3 ou mais avaliadores (McHugh 2012). Para realizar o cálculo do valor do *kappa*, foi utilizado um *script* em R (Anexo IV).

Utilizando a função “*kappam.fleiss*” com *um data frame* semelhante à tabela apresentada no Anexo III como parâmetro, é calculado um *kappa* de -0.066.

Também foi calculada uma métrica mais simples que representa a percentagem de concordância, que calcula o número de vezes que os investigadores deram a mesma resposta numa questão. Neste caso foi utilizada a função “*agree*” com os mesmos dados, tendo resultado num valor igual a 0%.

### 5.5.3 Conclusões

Começando pela concordância entre avaliadores, com um *kappa* de -0.066 e uma percentagem de concordância de 0%, é claro que em termos de concordância entre os diferentes investigadores envolvidos no inquérito esta é inexistente. Esta não é uma conclusão positiva, mas não reflete a sua opinião em relação ao componente de consulta de informação agregada e solução em geral.

Considerando o que foi mencionado na secção Inquérito de Satisfação, verifica-se que no mínimo 50% dos investigadores indicaram estar muito satisfeitos com o tópico abordado por

cada questão. Sendo assim, é possível concluir que o componente de consulta de informação agregada e *dashboards* são capazes de apresentar informação de uma forma precisa e compreensível para o utilizador. No entanto pode ainda ser melhorado especialmente na apresentação de informação considerada necessária, devido às respostas da Figura 85.

No caso da solução em geral, é possível concluir que esta vai de encontro às necessidades do cliente, facilitando-lhe diferentes aspetos, mas ainda pode ser melhorada.

## 6 Conclusão

Neste capítulo são apresentadas as conclusões atingidas nesta dissertação. Primeiro é realizada uma apreciação do trabalho desenvolvido, seguida da apresentação das suas limitações e de algumas considerações sobre o trabalho futuro.

### 6.1 Apreciação Final

No GECAD a gestão dos vários projetos em desenvolvimento e dos seus diferentes aspetos, é uma tarefa morosa. Este problema deu origem ao desenvolvimento de um portal denominado GesProj, através de um projeto de PESTI, com o objetivo de facilitar a gestão de projetos.

Enquanto que o GesProj permitia facilitar aspetos relacionados com a gestão de projetos, recursos humanos e aspetos financeiros, a alocação de investigadores a projetos ainda tinha de ser realizada manualmente. Enquanto que a consulta de informação de uma forma agregada tinha de ser realizada através de folhas Excel.

Ou seja, o GesProj apresentava limitações que podiam ser melhoradas. Primeiro para o processo de alocação de investigadores, foi identificada uma oportunidade para a criação de um sistema de recomendação de investigadores que facilite a criação de equipas no momento de elaboração de candidaturas a novos projetos. No caso da consulta de informação, existia a necessidade de algo que agregasse a informação contida no sistema, neste caso através *dashboards*.

Tendo estabelecido o problema e objetivos da dissertação, foi realizado um estudo relativamente às abordagens existentes de sistemas de recomendação, tanto de uma perspetiva geral e de uma perspetiva mais focada na gestão de projetos e recursos humanos. Através deste estudo foi possível determinar quais abordagens utilizar e que características seriam desejáveis na criação de um sistema de recomendação.

Também foi realizada a análise de valor, onde foi determinado o valor percebido e proposta de valor da solução, com o objetivo de determinar as vantagens que justificariam o seu desenvolvimento do ponto de vista do cliente.

Após o estudo do estado da arte e realização da análise de valor foi desenhada a solução, tendo-se optado por uma abordagem modular onde ao GesProj foi adicionado um componente de

recomendação e outro de consulta de informação agregada. Estes interagem diretamente com a base de dados do sistema e são acedidos diretamente pela aplicação *front-end* do GesProj.

Com base na avaliação das abordagens de recomendação analisadas no documento, decidiu-se que a melhor abordagem a implementar seria uma abordagem híbrida, composta por uma abordagem colaborativa seguida de uma abordagem baseada em conteúdo. Isto implicou a implementação da lógica destas duas abordagens antes de se implementar a abordagem híbrida. Também se estabeleceram novos conceitos de domínio que afetariam o funcionamento do GesProj, para lá da implementação das recomendações e *dashboards*.

Com o *design* concluído, foi implementada a solução. O que envolveu alterações no GesProj existente para este lidar com os novos conceitos de domínio, como Áreas, a criação do componente de consulta de informação e respetivas *dashboards*, implementadas através da biblioteca JavaScript amCharts. No caso do componente de recomendação foram desenvolvidas cinco abordagens distintas, uma com base em numa abordagem baseada em conteúdo, duas com base em recomendações colaborativas, dando origem assim a duas abordagens híbridas.

Com a solução desenvolvida esta foi avaliada. Para avaliar o sistema de recomendação foram estudadas as Métricas de Precisão do Apoio à Decisão (Precisão, *Recall*, *F-Measure*) com base em cinco cenários de execução diferentes. Para os valores obtidos foram criados *box-plots* através dos quais se deduziu que a melhor abordagem seria a A1, seguido da aplicação do teste de Friedman que permitiu concluir que existiam diferenças no desempenho de cada abordagem. Para verificar que a abordagem A1 seria de facto a melhor utilizou-se *Wilcoxon signed rank test* para a comparar às restantes, no entanto não foi possível através deste método concluir que a abordagem tem de facto o melhor desempenho.

Sendo assim utilizou-se a tabela de resultados (Tabela 16) e os *box-plots* criados para deduzir e concluir qual seria de facto a melhor abordagem. Assim deduziu-se que num cenário onde nenhum investigador está afeto a um projeto qualquer abordagem irá apresentar resultados que vão de encontro às preferências do utilizador. No caso de um período de recomendação onde os investigadores apenas estejam afetos durante uma porção do tempo indicado, a melhor abordagem seria a A1. No caso de um período onde a maioria dos investigadores já se encontra afeto a projetos, a melhor abordagem seria a A4. Tendo escolhido as abordagens A1 e A4 como as melhores, através das médias dos seus *F-Measures*, conclui-se que de facto a melhor abordagem é a A1.

Para avaliar o componente de consulta de informação e a solução em geral, foi utilizado um inquérito respondido por 8 investigadores do GECAD. Com base nas suas respostas foi possível concluir que para cada tópico abordado pelas questões, no mínimo 50% dos investigadores se encontrava muito satisfeito com este. Logo foi possível concluir que a maioria dos investigadores estavam muito satisfeitos com a solução desenvolvida, mesmo que vejam nesta, aspetos que possam ser melhorados.

## 6.2 Limitações e Trabalho futuro

Parte das limitações e trabalho futuro implicado nesta solução já estavam presentes na solução original, como é referido na secção GesProj. Pois ainda não é possível gerar folhas de pagamento, assim como ainda não foi implementado um sistema de *login* que os utilizadores possam utilizar. Estes aspetos não foram implementados pois não eram o foco deste projeto, mas merecem mesmo assim ser mencionados como o foco de um possível trabalho futuro.

Passando ao que foi desenvolvido é importante mencionar que a *timeline*, enquanto capaz de apresentar a informação necessária e transmitir o estado de um projeto, não apresenta alertas em caso de situações irregulares. Permitindo facilitar o processo de identificação de tais situações por parte dos utilizadores, mas não as identifica nem as apresenta aos utilizadores. De uma forma geral, segundo as respostas ao inquérito de avaliação, as *dashboards* beneficiariam de apresentar mais informação considerada necessária.

Relativamente à relação entre uma Atividade e uma Área, existe a possibilidade de em vez de uma atividade ter uma *tag*, ter várias. Tal alteração não só aumentaria a capacidade de caracterizar uma atividade, mas também aumentaria a experiência que cada investigador teria em cada área se uma atividade tivesse mais que uma área associada.

Mencionando a experiência de um investigador, de momento o seu perfil científico tem de ser gerido manualmente. No entanto, existiriam benefícios na implementação de algum método que consiga deduzir e atualizar o perfil do investigador, a partir de informação do sistema ou até informação exterior, como por exemplo alterando o seu curriculum através de plataformas como o ORCID<sup>1</sup> ou *cienciavita*<sup>2</sup>.

---

<sup>1</sup> <https://orcid.org>

<sup>2</sup> <https://cienciavita.pt>

Quanto às abordagens de recomendação implementadas, estas são apenas cinco das várias possíveis devido a grande número de abordagens de recomendação existente e formas de as implementar.

# Bibliografia

- Adomavicius, Gediminas, Nikos Manouselis, e YoungOk Kwon. 2011. «Multi-Criteria Recommender Systems». : 34.
- Agarwal, Ajay, e Minakshi Chauhan. 2017. «Similarity Measures Used in Recommender Systems: A Study». 4(6): 8.
- Aggarwal, Charu C. 2016a. «An Introduction to Recommender Systems». Em *Recommender Systems: The Textbook*, ed. Charu C. Aggarwal. Cham: Springer International Publishing, 1–28. [https://doi.org/10.1007/978-3-319-29659-3\\_1](https://doi.org/10.1007/978-3-319-29659-3_1) (21 de Outubro de 2019).
- . 2016b. «Content-Based Recommender Systems». Em *Recommender Systems: The Textbook*, ed. Charu C. Aggarwal. Cham: Springer International Publishing, 139–66. [https://doi.org/10.1007/978-3-319-29659-3\\_4](https://doi.org/10.1007/978-3-319-29659-3_4) (21 de Outubro de 2019).
- . 2016c. «Ensemble-Based and Hybrid Recommender Systems». Em *Recommender Systems: The Textbook*, ed. Charu C. Aggarwal. Cham: Springer International Publishing, 199–224. [https://doi.org/10.1007/978-3-319-29659-3\\_6](https://doi.org/10.1007/978-3-319-29659-3_6) (21 de Outubro de 2019).
- Alkan, Ozgur, Elizabeth M. Daly, e Inge Vejsbjerg. 2018. «Opportunity Team Builder for Sales Teams». Em *23rd International Conference on Intelligent User Interfaces, IUI '18*, New York, NY, USA: Association for Computing Machinery, 251–261. <https://doi.org/10.1145/3172944.3172968> (25 de Setembro de 2020).
- azizkhan. 2012. «Quality Function Deployment- Remote Control Example». <https://www.coursehero.com/file/18399429/CLASS-NOTES-02-03-04/>.
- Beklemysheva, Angela. 2018. «Why Use Python for AI and Machine Learning?» <https://steelkiwi.com/blog/python-for-ai-and-machine-learning/> (25 de Setembro de 2020).
- Bell, Suzanne T., Shanique G. Brown, e Jake A. Weiss. 2018. «A Conceptual Framework for Leveraging Team Composition Decisions to Build Human Capital». *Human Resource Management Review* 28(4): 450–63.
- bricelam. 2018. «Migrations - EF Core». *Migrations*. <https://docs.microsoft.com/en-us/ef/core/managing-schemas/migrations/> (16 de Maio de 2020).
- Chan, Lai-kow, e Ming-lu Wu. 2001. *Invited Review Quality function deployment: A literature review*.

- Clayton, Scott. 2018. «Building a Recommendation Engine in C#». <https://www.codeproject.com/Articles/1232150/Building-a-Recommendation-Engine-in-Csharp> (22 de Junho de 2020).
- Colomo Palacios, Ricardo, Israel González Carrasco, José Luis López Cuadrado, e Ángel García Crespo. 2012. «ReSySTER: A Hybrid Recommender System for Scrum Team Roles Based on Fuzzy and Rough Sets». *International Journal of Applied Mathematics and Computer Science* 22(4): 801–16.
- Couto, João. 2019. *Plataforma para Gestão de Recursos Humanos em Projetos de Investigação Científica*.
- Datta, Anwitaman, Stefano Braghin, e Jackson Tan Teck Yong. 2013. «The Zen of Multidisciplinary Team Recommendation». *arXiv:1303.0646 [physics]*. <http://arxiv.org/abs/1303.0646> (8 de Janeiro de 2020).
- Eeles, Peter. 2001. «Capturing Architectural Requirements».
- Facebook. 2019. «GitHub - facebook/react: A declarative, efficient, and flexible JavaScript library for building user interfaces.» <https://github.com/facebook/react> (9 de Dezembro de 2019).
- Farragher, Mark. 2019. «Build A Product Recommender Using C# and ML.NET Machine Learning». *Medium*. <https://medium.com/machinelearningadvantage/build-a-product-recommender-using-c-and-ml-net-machine-learning-ab890b802d25> (19 de Junho de 2020).
- Google. 2018. «Classification: Precision and Recall | Machine Learning Crash Course». *Classification: Precision and Recall*. <https://developers.google.com/machine-learning/crash-course/classification/precision-and-recall> (15 de Fevereiro de 2020).
- Kazmi, Najeeb. 2020. «Dotnet/Machinelearning». *GitHub*. <https://github.com/dotnet/machinelearning>.
- Kim, Kwang-Jae, Herbert Moskowitz, Anoop Dhingra, e Gerald Evans. 2000. «Fuzzy Multicriteria Models for Quality Function Deployment». *European Journal of Operational Research* 121(3): 504–18.
- Kluver, Daniel, Michael D. Ekstrand, e Joseph A. Konstan. 2018. «Rating-Based Collaborative Filtering: Algorithms and Evaluation». Em *Social Information Access: Systems and Technologies*, Lecture Notes in Computer Science, eds. Peter Brusilovsky e Daqing He. Cham: Springer International Publishing, 344–90. [https://doi.org/10.1007/978-3-319-90092-6\\_10](https://doi.org/10.1007/978-3-319-90092-6_10) (21 de Outubro de 2019).

- Koen, Peter et al. 2001. «Providing clarity and a common language to the “fuzzy front end”». *Research-Technology Management* 44(2): 46–55.
- Koen, Peter A. et al. 2002. «Fuzzy front end: effective methods, tools, and techniques». *The PDMA toolbox 1 for new product development*.
- Lu, Jie et al. 2015. «Recommender system application developments: A survey». *Decision Support Systems* 74: 12–32.
- Majeris, Martynas. 2020. *amcharts/amcharts4*. TypeScript. <https://github.com/amcharts/amcharts4> (20 de Junho de 2020).
- Malinowski, Jochen, Tim Weitzel, e Tobias Keim. 2008. «Decision Support for Team Staffing: An Automated Relational Recommendation Approach». *Decision Support Systems* 45(3): 429–47.
- McCrum-Gardner, Evie. 2008. «Which Is the Correct Statistical Test to Use?». *British Journal of Oral and Maxillofacial Surgery* 46(1): 38–41.
- McHugh, Mary L. 2012. «Interrater Reliability: The Kappa Statistic». *Biochemia Medica* 22(3): 276–82.
- Nicola, Susana. 2019. «Análise de Valor».
- Olszak, Celina M. 2016. «Toward Better Understanding and Use of Business Intelligence in Organizations». *Information Systems Management* 33(2): 105–23.
- Osterwalder, Alexander, e Yves Pigneur. 2010. «You’re Holding a Handbook for Visionaries, Game Changers, and Challengers Striving to Defy Outmoded Business Models and Design Tomorrow’s Enterprises. It’s a Book for The...». : 51.
- Osterwalder, Alexander, Yves Pigneur, Gregory Bernarda, e Alan Smith. 2014. *Value Proposition Design: How to Create Products and Services Customers Want*. John Wiley & Sons.
- Quintanilla, Luis. 2019. «What Is ML.NET and How Does It Work? - ML.NET». *What is ML.NET and how does it work?* <https://docs.microsoft.com/en-us/dotnet/machine-learning/how-does-mldotnet-work> (19 de Junho de 2020).
- Roth, Daniel, Rick Anderson, e Luttin Shaun. 2019. «Introduction to ASP.NET Core | Microsoft Docs». <https://docs.microsoft.com/en-us/aspnet/core/?view=aspnetcore-3.1> (9 de Dezembro de 2019).
- Saaty, Thomas L. 1988. «What Is the Analytic Hierarchy Process?». Em *Mathematical Models for Decision Support*, eds. Gautam Mitra et al. Berlin, Heidelberg: Springer Berlin

- Heidelberg, 109–21. [http://link.springer.com/10.1007/978-3-642-83555-1\\_5](http://link.springer.com/10.1007/978-3-642-83555-1_5) (4 de Dezembro de 2019).
- . 1990. «How to Make a Decision: The Analytic Hierarchy Process». *European Journal of Operational Research* 48(1): 9–26.
- Sarwar, Badrul, George Karypis, Joseph Konstan, e John Reidl. 2001. «Item-Based Collaborative Filtering Recommendation Algorithms». Em *Proceedings of the Tenth International Conference on World Wide Web - WWW '01*, Hong Kong, Hong Kong: ACM Press, 285–95. <http://portal.acm.org/citation.cfm?doid=371920.372071> (23 de Dezembro de 2019).
- Singh, Suraj Pal, e Shano Solanki. 2019. «Recommender System Survey: Clustering to Nature Inspired Algorithm». Em *Proceedings of 2nd International Conference on Communication, Computing and Networking*, Lecture Notes in Networks and Systems, eds. C. Rama Krishna, Maitreyee Dutta, e Rakesh Kumar. Springer Singapore, 757–68.
- Sweeney, Jillian C., e Geoffrey N. Soutar. 2001. «Consumer perceived value: The development of a multiple item scale - ScienceDirect». <https://www.sciencedirect.com/science/article/abs/pii/S0022435901000410> (2 de Dezembro de 2019).
- TechMagic. 2018. «React vs Angular vs Vue.js — What to choose in 2019? (updated)». <https://medium.com/@TechMagic/reactjs-vs-angular5-vs-vue-js-what-to-choose-in-2018-b91e028fa91d> (9 de Dezembro de 2019).
- Varga, Ervin. 2019. «Recommender Systems». Em *Practical Data Science with Python 3: Synthesizing Actionable Insights from Data*, ed. Ervin Varga. Berkeley, CA: Apress, 317–39. [https://doi.org/10.1007/978-1-4842-4859-1\\_8](https://doi.org/10.1007/978-1-4842-4859-1_8) (16 de Outubro de 2019).
- Warren, Genevieve et al. 2019. «Tutorial: Build a movie recommender - matrix factorization - ML.NET | Microsoft Docs». <https://docs.microsoft.com/en-us/dotnet/machine-learning/tutorials/movie-recommendation> (9 de Dezembro de 2019).
- Zealous System. 2018. «Why Enterprises Are Turning to ASP.NET Core for Web Application Development». <https://codeburst.io/why-enterprises-are-turning-to-asp-net-core-for-web-application-development-8918cf6148c1> (9 de Dezembro de 2019).



# Folha de afetação de um projeto

		set-18	out-18	nov-18	dez-18	jan-19	fev-19	mar-19	abr-19	maio-19	jun-19	jul-19	ago-19	set-19	out-19	nov-19	dez-19	jan-20	fev-20	mar-20	abr-20	maio-20	jun-20	jul-20	ago-20	set-20	out-20	nov-20	dez-20	jan-21	fev-21	mar-21	abr-21	maio-21	jun-21	jul-21	ago-21		
	Total pre total aloc	0																																					
hml1	1	0																																					
	2	1,71	0,0725	0,0725	0,0725	0,0725	0,0725	0,0725	0,0725	0,0725	0,0725	0,0725	0,0725	0,0725	0,0725	0,0725	0,0725	0,0725	0,0725	0,0725	0,0725	0,0725	0,0725	0,0725	0,0725	0,0725	0,0725	0,0725	0,0725	0,0725	0,0725	0,0725	0,0725	0,0725	0,0725	0,0725	0,0725	0,0725	0,0725
	3	0,51	0,05	0,05	0,05	0,05	0,05	0,05	0,05	0,05	0,05	0,05	0,05	0,05	0,05	0,05	0,05	0,05	0,05	0,05	0,05	0,05	0,05	0,05	0,05	0,05	0,05	0,05	0,05	0,05	0,05	0,05	0,05	0,05	0,05	0,05	0,05	0,05	
	4	0																																					
	5	0																																					
	6	0																																					
	7	1,08	0,08	0,08	0,08	0,08	0,08	0,08	0,08	0,08	0,08	0,08	0,08	0,08	0,08	0,08	0,08	0,08	0,08	0,08	0,08	0,08	0,08	0,08	0,08	0,08	0,08	0,08	0,08	0,08	0,08	0,08	0,08	0,08	0,08	0,08	0,08	0,08	
	Total	3,3	0,1825	0,1825	0,1825	0,1825	0,1825	0,1825	0,1825	0,1825	0,1825	0,1825	0,1825	0,1825	0,1825	0,1825	0,1825	0,1825	0,1825	0,1825	0,1825	0,1825	0,1825	0,1825	0,1825	0,1825	0,1825	0,1825	0,1825	0,1825	0,1825	0,1825	0,1825	0,1825	0,1825	0,1825	0,1825	0,1825	
hml2	1	0																																					
	2	5,1	0,2125	0,2125	0,2125	0,2125	0,2125	0,2125	0,2125	0,2125	0,2125	0,2125	0,2125	0,2125	0,2125	0,2125	0,2125	0,2125	0,2125	0,2125	0,2125	0,2125	0,2125	0,2125	0,2125	0,2125	0,2125	0,2125	0,2125	0,2125	0,2125	0,2125	0,2125	0,2125	0,2125	0,2125	0,2125	0,2125	0,2125
	3	1,5	0,0418	0,0418	0,0418	0,0418	0,0418	0,0418	0,0418	0,0418	0,0418	0,0418	0,0418	0,0418	0,0418	0,0418	0,0418	0,0418	0,0418	0,0418	0,0418	0,0418	0,0418	0,0418	0,0418	0,0418	0,0418	0,0418	0,0418	0,0418	0,0418	0,0418	0,0418	0,0418	0,0418	0,0418	0,0418	0,0418	0,0418
	4	0																																					
	5	0																																					
	6	0																																					
	7	0																																					
	Total	6,6	0,2568	0,2568	0,2568	0,2568	0,2568	0,2568	0,2568	0,2568	0,2568	0,2568	0,2568	0,2568	0,2568	0,2568	0,2568	0,2568	0,2568	0,2568	0,2568	0,2568	0,2568	0,2568	0,2568	0,2568	0,2568	0,2568	0,2568	0,2568	0,2568	0,2568	0,2568	0,2568	0,2568	0,2568	0,2568	0,2568	0,2568
hml3	1	0																																					
	2	3,9	0,1625	0,1625	0,1625	0,1625	0,1625	0,1625	0,1625	0,1625	0,1625	0,1625	0,1625	0,1625	0,1625	0,1625	0,1625	0,1625	0,1625	0,1625	0,1625	0,1625	0,1625	0,1625	0,1625	0,1625	0,1625	0,1625	0,1625	0,1625	0,1625	0,1625	0,1625	0,1625	0,1625	0,1625	0,1625	0,1625	0,1625
	3	1,65	0,0629	0,0629	0,0629	0,0629	0,0629	0,0629	0,0629	0,0629	0,0629	0,0629	0,0629	0,0629	0,0629	0,0629	0,0629	0,0629	0,0629	0,0629	0,0629	0,0629	0,0629	0,0629	0,0629	0,0629	0,0629	0,0629	0,0629	0,0629	0,0629	0,0629	0,0629	0,0629	0,0629	0,0629	0,0629	0,0629	0,0629
	4	0																																					
	5	0																																					
	6	1,05	0,05																																				
	7	0																																					
	Total	6,6	0,2629	0,2629	0,2629	0,2629	0,2629	0,2629	0,2629	0,2629	0,2629	0,2629	0,2629	0,2629	0,2629	0,2629	0,2629	0,2629	0,2629	0,2629	0,2629	0,2629	0,2629	0,2629	0,2629	0,2629	0,2629	0,2629	0,2629	0,2629	0,2629	0,2629	0,2629	0,2629	0,2629	0,2629	0,2629	0,2629	0,2629

## Anexo II – Resultados dos Cenários de Avaliação

	Cenário 1	Cenário 2	Cenário 3	Cenário 4	Cenário 5
A1	Inv6, Inv1, Inv5, Inv3, Inv2	Inv6, Inv1, Inv5, Inv3, Inv2	Inv6, Inv5, Inv1, Inv2, Inv3	Inv6, Inv1, Inv4, Inv5, Inv3, Inv2	Inv6, Inv5, Inv2, Inv3, Inv4, Inv1
A2	Inv1, Inv5, Inv4, Inv3, Inv2	Inv1, Inv4, Inv5, Inv3, Inv2	Inv1, Inv4, Inv5, Inv3, Inv2	Inv1, Inv4, Inv5, Inv3, Inv2	Inv2, Inv5, Inv1, Inv4, Inv6, Inv3
A3	Inv4, Inv1, Inv2, Inv3, Inv5	Inv1, Inv4, Inv2, Inv3	Inv4, Inv1, Inv2, Inv3, Inv5	Inv1, Inv4, Inv2, Inv3, Inv5	Inv2, Inv3, Inv4, Inv5, Inv1, Inv6
A4	Inv1, Inv5, Inv3, Inv2	Inv1, Inv5, Inv4, Inv3, Inv2	Inv5, Inv1, Inv4, Inv3, Inv6, Inv2	Inv1, Inv4, Inv5, Inv3, Inv6, Inv2	Inv6, Inv5, Inv2, Inv4, Inv1, Inv3
A5	Inv1, Inv3, Inv2, Inv5	Inv1, Inv4, Inv3, Inv5, Inv2	Inv5, Inv1, Inv4, Inv3, Inv6, Inv2	Inv1, Inv4, Inv5, Inv3, Inv2	Inv5, Inv2, Inv3, Inv4, Inv1, Inv6



## Anexo III – Respostas ao Inquérito

Aqui são apresentadas as respostas ao inquérito por investigador. Por motivos de simplicidade cada pergunta é referida como Q1, Q2 e assim sucessivamente. Os investigadores mantêm o formato estabelecido anteriormente Inv1, Inv2 e assim sucessivamente. As respostas também são simplificadas quanto à sua representação para facilitar a leitura da tabela.

- Muito insatisfeito (MI).
- Insatisfeito (I).
- Sem opinião (SO).
- Satisfeito (S).
- Muito satisfeito (MS).

Questões	Inv1	Inv2	Inv3	Inv4	Inv5	Inv6	Inv7	Inv8
Q1	MS	S	MS	MS	S	S	MS	MS
Q2	MS	MS	SO	S	S	S	MS	MS
Q3	MS	MS	SO	SO	S	S	MS	MS
Q4	MS	SO	MS	MS	SO	S	MS	S
Q5	MS	SO	MS	S	MS	S	MS	MS
Q6	MS	SO	MS	MS	MS	S	MS	MS
Q7	MS	MS	MS	MS	MS	S	MS	MS



# Anexo IV – Scripts R

## Sistema de Recomendações

```
div <- function(v1,v2){
  r <- round(v1/v2,digits = 4)
  return (r)
}

fmeasure <- function(v1,v2){
  f <- round((2*v1*v2)/(v1+v2),digits = 4)
  return (f)
}

library(tidyverse)
library(ggpubr)
library(rstatix)
library(car)
library(reshape2)
library(dplyr)

#-----
#A1

a1pc1 = div(4,5)
a1rc1 = div(4,4)
a1fmc1 = fmeasure(a1pc1,a1rc1)

a1pc2 = div(4,5)
a1rc2 = div(4,4)
a1fmc2 = fmeasure(a1pc2,a1rc2)

a1pc3 = div(2,5)
a1rc3 = div(2,3)
a1fmc3 = fmeasure(a1pc3,a1rc3)

a1pc4 = div(3,6)
```

```

a1rc4 = div(3,3)
a1fmc4 = fmeasure(a1pc4,a1rc4)

a1pc5 = div(6,6)
a1rc5 = div(6,6)
a1fmc5 = fmeasure(a1pc5,a1rc5)

cat(paste("A1 - C1 -",a1pc1,"-",a1rc1,"-",a1fmc1))
cat(paste("A1 - C2 -",a1pc2,"-",a1rc2,"-",a1fmc2))
cat(paste("A1 - C3 -",a1pc3,"-",a1rc3,"-",a1fmc3))
cat(paste("A1 - C4 -",a1pc4,"-",a1rc4,"-",a1fmc4))
cat(paste("A1 - C5 -",a1pc5,"-",a1rc5,"-",a1fmc5))

```

#A2

```

a2pc1 = div(3,5)
a2rc1 = div(3,4)
a2fmc1 = fmeasure(a2pc1,a2rc1)

```

```

a2pc2 = div(3,5)
a2rc2 = div(3,4)
a2fmc2 = fmeasure(a2pc2,a2rc2)

```

```

a2pc3 = div(2,5)
a2rc3 = div(2,3)
a2fmc3 = fmeasure(a2pc3,a2rc3)

```

```

a2pc4 = div(2,5)
a2rc4 = div(2,3)
a2fmc4 = fmeasure(a2pc4,a2rc4)

```

```

a2pc5 = div(6,6)
a2rc5 = div(6,6)
a2fmc5 = fmeasure(a2pc5,a2rc5)

```

```

cat(paste("A2 - C1 -",a2pc1,"-",a2rc1,"-",a2fmc1))
cat(paste("A2 - C2 -",a2pc2,"-",a2rc2,"-",a2fmc2))
cat(paste("A2 - C3 -",a2pc3,"-",a2rc3,"-",a2fmc3))
cat(paste("A2 - C4 -",a2pc4,"-",a2rc4,"-",a2fmc4))
cat(paste("A2 - C5 -",a2pc5,"-",a2rc5,"-",a2fmc5))

```

```

#A3

a3pc1 = div(3,5)
a3rc1 = div(3,4)
a3fmc1 = fmeasure(a3pc1,a3rc1)

a3pc2 = div(2,4)
a3rc2 = div(2,4)
a3fmc2 = fmeasure(a3pc2,a3rc2)

a3pc3 = div(2,5)
a3rc3 = div(2,3)
a3fmc3 = fmeasure(a3pc3,a3rc3)

a3pc4 = div(2,5)
a3rc4 = div(2,3)
a3fmc4 = fmeasure(a3pc4,a3rc4)

a3pc5 = div(6,6)
a3rc5 = div(6,6)
a3fmc5 = fmeasure(a3pc5,a3rc5)

cat(paste("A3 - C1 -",a3pc1,"-",a3rc1,"-",a3fmc1))
cat(paste("A3 - C2 -",a3pc2,"-",a3rc2,"-",a3fmc2))
cat(paste("A3 - C3 -",a3pc3,"-",a3rc3,"-",a3fmc3))
cat(paste("A3 - C4 -",a3pc4,"-",a3rc4,"-",a3fmc4))
cat(paste("A3 - C5 -",a3pc5,"-",a3rc5,"-",a3fmc5))

#A4

a4pc1 = div(3,4)
a4rc1 = div(3,4)
a4fmc1 = fmeasure(a4pc1,a4rc1)

a4pc2 = div(3,5)
a4rc2 = div(3,4)
a4fmc2 = fmeasure(a4pc2,a4rc2)

a4pc3 = div(3,6)

```

```

a4rc3 = div(3,3)
a4fmc3 = fmeasure(a4pc3,a4rc3)

a4pc4 = div(3,6)
a4rc4 = div(3,3)
a4fmc4 = fmeasure(a4pc4,a4rc4)

a4pc5 = div(6,6)
a4rc5 = div(6,6)
a4fmc5 = fmeasure(a4pc5,a4rc5)

cat(paste("A4 - C1 -",a4pc1,"-",a4rc1,"-",a4fmc1))
cat(paste("A4 - C2 -",a4pc2,"-",a4rc2,"-",a4fmc2))
cat(paste("A4 - C3 -",a4pc3,"-",a4rc3,"-",a4fmc3))
cat(paste("A4 - C4 -",a4pc4,"-",a4rc4,"-",a4fmc4))
cat(paste("A4 - C5 -",a4pc5,"-",a4rc5,"-",a4fmc5))

#A5

a5pc1 = div(3,4)
a5rc1 = div(3,4)
a5fmc1 = fmeasure(a5pc1,a5rc1)

a5pc2 = div(3,5)
a5rc2 = div(3,4)
a5fmc2 = fmeasure(a5pc2,a5rc2)

a5pc3 = div(3,6)
a5rc3 = div(3,3)
a5fmc3 = fmeasure(a5pc3,a5rc3)

a5pc4 = div(2,5)
a5rc4 = div(2,3)
a5fmc4 = fmeasure(a5pc4,a5rc4)

a5pc5 = div(6,6)
a5rc5 = div(6,6)
a5fmc5 = fmeasure(a5pc5,a5rc5)

cat(paste("A5 - C1 -",a5pc1,"-",a5rc1,"-",a5fmc1))

```

```

cat(paste("A5 - C2 -", a5pc2, "-", a5rc2, "-", a5fmc2))
cat(paste("A5 - C3 -", a5pc3, "-", a5rc3, "-", a5fmc3))
cat(paste("A5 - C4 -", a5pc4, "-", a5rc4, "-", a5fmc4))
cat(paste("A5 - C5 -", a5pc5, "-", a5rc5, "-", a5fmc5))

#Precisao = N de itens relevantes recomendados /N de itens recomondados
#Recall = N de itens relevantes recomendados / N total de itens
relevantes
#F-Measure = 2xPrecisaoxRecall / (Precisao+Recall)

abordagens=c("A1", "A2", "A3", "A4", "A5")
cenarios=c("c1", "c2", "c3", "c4", "c5")

#BoxPlots-----

boxData1 = data.frame(a1=c(a1pc1, a1pc2, a1pc3, a1pc4, a1pc5),
                      a2=c(a2pc1, a2pc2, a2pc3, a2pc4, a2pc5),
                      a3=c(a3pc1, a3pc2, a3pc3, a3pc4, a3pc5),
                      a4=c(a4pc1, a4pc2, a4pc3, a4pc4, a4pc5),
                      a5=c(a5pc1, a5pc2, a5pc3, a5pc4, a5pc5))

boxplot(boxData1,
        main= "Precisão das Abordagens",
        names = abordagens,
        ylab="Precisão")

boxData2 = data.frame(a1=c(a1rc1, a1rc2, a1rc3, a1rc4, a1rc5),
                      a2=c(a2rc1, a2rc2, a2rc3, a2rc4, a2rc5),
                      a3=c(a3rc1, a3rc2, a3rc3, a3rc4, a3rc5),
                      a4=c(a4rc1, a4rc2, a4rc3, a4rc4, a4rc5),
                      a5=c(a5rc1, a5rc2, a5rc3, a5rc4, a5rc5))

boxplot(boxData2,
        main= "Recall das Abordagens",
        names = abordagens,
        ylab="Recall")

boxData3 = data.frame(a1=c(a1fmc1, a1fmc2, a1fmc3, a1fmc4, a1fmc5),
                      a2=c(a2fmc1, a2fmc2, a2fmc3, a2fmc4, a2fmc5),
                      a3=c(a3fmc1, a3fmc2, a3fmc3, a3fmc4, a3fmc5),

```

```

a4=c(a4fmc1,a4fmc2,a4fmc3,a4fmc4,a4fmc5),
a5=c(a5fmc1,a5fmc2,a5fmc3,a5fmc4,a5fmc5))

boxplot(boxData3,
        main= "F-Measure das Abordagens",
        names = abordagens,
        ylab="F-Measure")

#Tests-----

data = data.frame(cenarios,
                  A1=c(a1fmc1,a1fmc2,a1fmc3,a1fmc4,a1fmc5),
                  A2=c(a2fmc1,a2fmc2,a2fmc3,a2fmc4,a2fmc5),
                  A3=c(a3fmc1,a3fmc2,a3fmc3,a3fmc4,a3fmc5),
                  A4=c(a4fmc1,a4fmc2,a4fmc3,a4fmc4,a4fmc5),
                  A5=c(a5fmc1,a5fmc2,a5fmc3,a5fmc4,a5fmc5))

fData <- data %>%
  gather(key = "abordagem", value = "valor", A1, A2, A3, A4, A5) %>%
  convert_as_factor(cenarios, valor)

fData$valor <- as.numeric(as.character(fData$valor))

fData %>%
  group_by(abordagem) %>%
  get_summary_stats(valor, type = "common")

res.fried <- fData %>% friedman_test(valor ~ abordagem |cenarios)
res.fried

wilcox.test(data$A1,
            data$A2,
            paired = TRUE,
            conf.level = 0.95,
            alternative = "greater")
wilcox.test(data$A1,
            data$A3,
            paired = TRUE,
            conf.level = 0.95,
            alternative = "greater")

```

```
wilcox.test(data$A1,
            data$A4,
            paired = TRUE,
            conf.level = 0.95,
            alternative = "greater")
wilcox.test(data$A1,
            data$A5,
            paired = TRUE,
            conf.level = 0.95,
            alternative = "greater")
```

## ***Dashboards***

```
library(RColorBrewer)
library(lpSolve)
library(irr)
library(tidyverse)

data("diagnoses", package = "irr")
kappam.fleiss(diagnoses, detail = TRUE)

baseAnswers = c("Muito insatisfeito", "Insatisfeito", "Sem
opinioao", "Satisfeito", "Muito satisfeito")

#Concordancia-----
answers = data.frame(rater1=c("Muito satisfeito", "Muito
satisfeito", "Muito satisfeito", "Muito satisfeito", "Muito
satisfeito", "Muito satisfeito", "Muito satisfeito"),
                    rater2=c("Satisfeito", "Muito satisfeito", "Muito
satisfeito", "Sem opiniao", "Sem opiniao", "Sem opiniao", "Muito
satisfeito"),
                    rater3=c("Muito satisfeito", "Sem opiniao", "Sem
opinioao", "Muito satisfeito", "Muito satisfeito", "Muito satisfeito", "Muito
satisfeito"),
                    rater4=c("Muito satisfeito", "Satisfeito", "Sem
opinioao", "Muito satisfeito", "Satisfeito", "Muito satisfeito", "Muito
satisfeito"),
                    rater5=c("Satisfeito", "Satisfeito", "Satisfeito", "Sem opiniao", "Muito
satisfeito", "Muito satisfeito", "Muito satisfeito"),
                    rater6=c("Satisfeito", "Satisfeito", "Satisfeito", "Satisfeito", "Satisfeito
", "Satisfeito", "Satisfeito"),
                    rater7=c("Muito satisfeito", "Muito
satisfeito", "Muito satisfeito", "Muito satisfeito", "Muito
satisfeito", "Muito satisfeito", "Muito satisfeito"),
```

```
      rater8=c("Muito satisfeito", "Muito
satisfeito", "Muito satisfeito", "Satisfeito", "Muito satisfeito", "Muito
satisfeito", "Muito satisfeito"))
```

```
answers$rater1 <- as.factor(answers$rater1)
answers$rater2 <- as.factor(answers$rater2)
answers$rater3 <- as.factor(answers$rater3)
answers$rater4 <- as.factor(answers$rater4)
answers$rater5 <- as.factor(answers$rater5)
answers$rater6 <- as.factor(answers$rater6)
answers$rater7 <- as.factor(answers$rater7)
answers$rater8 <- as.factor(answers$rater8)
```

```
kappam.fleiss(answers, detail = TRUE)
agree(answers)
```

```
#Graficos-----
```

```
color_range<- c(c1="#3366cc", c2="#dc3912", c3="#ff9900",
c4="#109618", c5="#990099")
```

```
barplot(c(0,0,0,3,5),
        col= color_range[1:5],
        names = baseAnswers,
        main = "A solução é simples e intuitiva de utilizar?")
```

```
barplot(c(0,0,1,3,4),
        col= color_range[1:5],
        names = baseAnswers,
        main = "O sistema de recomendação apresenta resultados úteis e
diversificados?")
```

```
barplot(c(0,0,2,2,4),
        col= color_range[1:5],
        names = baseAnswers,
        main = "A escolha de recursos humanos com base em recomendações
é simples?")
```

```
barplot(c(0,0,2,2,4),
        col= color_range[1:5],
        names = baseAnswers,
        main = "As dashboards apresentam toda a informação necessária?")
```

```
barplot(c(0,0,1,2,5),
        col= color_range[1:5],
        names = baseAnswers,
        main = "As dashboards apresentam informação de forma completa e
consistente?")
```

```
barplot(c(0,0,1,1,6),
        col= color_range[1:5],
        names = baseAnswers,
        main = "As dashboards apresentam informação de uma forma simples
de compreender?")
```

```
barplot(c(0,0,0,1,7),
        col= color_range[1:5],
        names = baseAnswers,
        main = "A solução vai de encontro às necessidades do cliente?")
```