

COIMBRA  
BUSINESS  
SCHOOL

 **iscac**   
Politécnico de Coimbra

**COIMBRA  
BUSINESS  
SCHOOL**  
 **iscac**   
Politécnico de Coimbra

Adinilsa Pontes

Composição de Equipas de Projetos  
de Desenvolvimento de *Software*  
com Inteligência Artificial

Coimbra, julho de 2024





Adinilsa Pontes

## Composição de Equipas de Projetos de Desenvolvimento de *Software* com Inteligência Artificial

Dissertação submetida ao Instituto Superior de Contabilidade e Administração de Coimbra para cumprimento dos requisitos necessários à obtenção do grau de Mestre em Sistema de Informação de Gestão, realizado sob a orientação do Professor Doutor António Trigo.

Coimbra, julho de 2024

## **TERMO DE RESPONSABILIDADE**

Declaro ser a autora desta dissertação, que constitui um trabalho original e inédito, que nunca foi submetido a outra Instituição de ensino superior para obtenção de um grau académico ou outra habilitação. Atesto ainda que todas as citações estão devidamente identificadas e que tenho consciência de que o plágio constitui uma grave falta de ética, que poderá resultar na anulação da presente dissertação.

## **PENSAMENTO**

*“Nenhum projeto é feito apenas de trabalho árduo e dedicação, mas também de apoio inestimável e amor incondicional. Que esta tese seja o reflexo do esforço conjunto, do apoio incansável e das aprendizagens valiosas que recebi ao longo do caminho. Que cada página escrita seja um tributo aos que estiveram ao meu lado, tornando este percurso possível”*

Adinlsa Dos Reis Sousa Pontes  
(nasceu a 21 de outubro de 1998 em  
São Tomé e Príncipe)

## **AGRADECIMENTOS**

Agradeço ao Instituto Politécnico de Coimbra, que não só me proporcionou uma educação de excelência, mas também um ambiente de apoio e crescimento pessoal.

Ao meu orientador, Professor Doutor António Trigo, cuja orientação sábia e incentivo constante foram um farol no meio da tempestade académica.

À minha família, cujo amor incondicional e apoio incansável foram a âncora que me sustentou nos momentos mais desafiadores.

Aos amigos e colegas de jornada, pela companhia, pelo riso compartilhado e pelo apoio mútuo que tornaram esta jornada académica memorável.

Este trabalho é fruto não apenas do meu esforço, mas também do apoio e inspiração de cada um de vocês.

## RESUMO

Atualmente, a Inteligência Artificial (IA) é considerada como um dos temas mais discutidos a nível mundial, não sendo de estranhar a sua utilização por parte das empresas de desenvolvimento de software na gestão dos seus projetos. A IA oferece ferramentas avançadas para automação de tarefas repetitivas, análise preditiva e otimização de processos, permitindo uma melhor gestão dos projetos de desenvolvimento de software. Estas tecnologias podem ajudar na deteção precoce de falhas, previsão de prazos e orçamentos, além de facilitar a comunicação e a colaboração entre equipas e de apoiar a tomada de decisão do gestor de projetos.

Neste contexto, o tema proposto explora a utilização da IA para composição de equipas de desenvolvimento de *software* de forma mais eficiente e eficaz, tendo sido desenvolvidos dois algoritmos de IA de *content matching*: O primeiro algoritmo procura identificar através da descrição do projeto quais os colaboradores com as competências mais adequadas para participar no projeto; e o segundo, compara a descrição do projeto com projetos passados, permitindo ao gestor de projetos ver quais os projetos passados mais similares com o projeto que pretende desenvolver, identificando o grau de sucesso e as equipas participantes, apoiando o gestor de projeto na tomada de decisão na seleção dos colaboradores para a equipa do projeto.

Através da revisão sistemática da literatura e da análise de casos práticos, foram identificados os principais benefícios e desafios da utilização de IA na composição de equipas, bem como as tendências emergentes neste campo. Os resultados demonstram que a IA tem o potencial de transformar a forma como as equipas de desenvolvimento de *software* são formadas, contribuindo significativamente para a eficiência e o sucesso dos projetos.

*Palavras-chave: Gestão de Projetos; Composição de Equipas; Inteligência Artificial; Content Matching; Similaridade.*

## **ABSTRACT**

Artificial Intelligence (AI) is currently considered to be one of the most discussed topics worldwide, and it's not surprising that software development companies are using it to manage their projects. AI offers advanced tools for automating repetitive tasks, predictive analysis and process optimization, enabling better management of software development projects. These technologies can help with early fault detection, forecasting deadlines and budgets, as well as facilitating communication and collaboration between teams and supporting the project manager's decision-making.

In this context, the proposed topic explores the use of AI to compose *software* development teams more efficiently and effectively, and two content matching AI algorithms have been developed: The first algorithm seeks to identify through the project description which employees have the most suitable skills to participate in the project; and the second, compares the project description with past projects, allowing the project manager to see which past projects are similar to the project they intend to develop, identifying the degree of success and the participating teams, supporting the project manager in making decisions when selecting employees for the project team.

By systematically reviewing the literature and analyzing case studies, the main benefits and challenges of using AI in team composition were identified, as well as emerging trends in this field. The results show that AI has the potential to transform the way *software* development teams are formed, contributing significantly to the efficiency and success of projects.

*Keywords: Project Management; Team Composition; Artificial Intelligence; Content Matching; Similarity.*

## ÍNDICE GERAL

INTRODUÇÃO .....	1
1 ENQUADRAMENTO TEÓRICO .....	4
1.1 Inteligência Artificial .....	4
1.2 Gestão de Projetos .....	6
1.3 Projetos de desenvolvimento de <i>software</i> .....	7
1.3.1 Tipos de projetos .....	8
1.3.2 Características dos colaboradores .....	8
1.4 Inteligência Artificial e a Gestão de Projetos .....	9
1.4.1 Algoritmos de Inteligência Artificial .....	11
1.4.2 Processamento de Linguagem Natural .....	12
2 REVISÃO DE LITERATURA .....	14
2.1 Planeamento .....	14
2.2 Execução .....	16
2.2.1 Identificar os estudos .....	16
2.2.2 Selecionar os estudos .....	17
2.2.3 Qualidade .....	17
2.2.4 Extrair os estudos .....	17
2.2.5 Síntese dos estudos .....	18
2.3 Comunicação .....	19
3 METODOLOGIA .....	25
4 EXPERIÊNCIA REALIZADA .....	27
4.1 Entendimento do negócio .....	27
4.2 Entendimento dos dados .....	27

*Composição de Equipas de Projetos de Desenvolvimento de Software com IA*

---

4.3	Preparação dos dados.....	28
4.3.1	Geração de dados sintéticos.....	28
4.3.2	Armazenamento dos dados gerados na base de dados.....	29
4.3.3	Pré-processamento de texto.....	31
4.4	Modelação.....	34
4.4.1	Métrica similaridade do cosseno.....	34
4.4.2	Algoritmo de <i>content matching</i> das competências.....	35
4.4.3	Algoritmo de <i>content matching</i> dos projetos.....	37
4.4.4	Aplicação gráfica de interface com o utilizador.....	38
4.5	Avaliação.....	42
4.5.1	Teste do algoritmo de <i>content matching</i> dos projetos.....	43
4.5.2	Teste do algoritmo de <i>content matching</i> das competências.....	44
4.6	Entrega.....	46
	CONCLUSÃO.....	47
	REFERÊNCIAS.....	49
	APÊNDICES.....	55
	APÊNDICE 1. Conjunto de dados Projeto.....	56
	APÊNDICE 2. Conjunto de dados Colaboradores.....	58
	ANEXOS.....	61
	ANEXO 1. Caso de Sucesso 1 da Empresa GESTP.....	62
	ANEXO 2. Caso de Sucesso 2 da Empresa GESTP.....	63

## **ÍNDICE DE TABELAS**

Tabela 1.1. Competências dos colaboradores das equipas de desenvolvimento de software .....	9
Tabela 1.2. Algoritmos de IA na gestão de projetos.....	11
Tabela 2.1. Lista de artigos resultante da RSL .....	18

## ÍNDICE DE FIGURAS

Figura 2.1. Rede bibliométrica obtida a partir da primeira expressão de pesquisa .....	15
Figura 2.2. Expressão de pesquisa da RSL .....	16
Figura 2.3. Área de pesquisa.....	20
Figura 2.4. Número de documentos por país .....	20
Figura 3.1. Modelo CRISP-DM.....	25
Figura 4.1. Extrato da tabela Projeto .....	30
Figura 4.2. Extrato da tabela Colaborador .....	30
Figura 4.3. Extrato da tabela Equipa.....	30
Figura 4.4. Modelo ER de suporte ao projeto criado na aplicação onda.dei.uc.pt .....	31
Figura 4.5. Extrato de código com a função de conexão à base de dados .....	31
Figura 4.6. Extrato de código Python com as bibliotecas importadas para o pré-processamento de texto .....	32
Figura 4.7. Texto após a tokenização .....	32
Figura 4.8. Texto após o pré-processamento .....	32
Figura 4.9. Método de pré- processamento.....	33
Figura 4.10. Pré-processamento de Texto para língua portuguesa .....	34
Figura 4.11. Exemplo do cálculo da métrica similaridade do cosseno em Python.....	35
Figura 4.12. Extrato do código responsável pelo cálculo da similaridade entre a descrição do projeto e das descrições das competências dos colaboradores .....	36
Figura 4.13. Resultado da comparação da descrição de um projeto com as competências dos colaboradores existentes na base de dados.....	36
Figura 4.14. Cálculo da similaridade entre descrições de projetos.....	37
Figura 4.15. Resultado da comparação da descrição de um projeto com os projetos existentes na base de dados.....	38

*Composição de Equipas de Projetos de Desenvolvimento de Software com IA*

---

Figura 4.16 Rota para processamento do formulário do algoritmo da comparação da descrição do projetos .....	39
Figura 4.17. Extrato do código do template para a apresentação do algoritmo de comparação da descrição dos projetos.....	40
Figura 4.18. Extrato do código do template para a apresentação do algoritmo de comparação da descrição dos projetos que contém a função de javascript de validação do formulário .....	41
Figura 4.19. Mensagem de aviso de descrição do projeto vazio .....	41
Figura 4.20. Função que emite a mensagem quando o projeto não é encontrado .....	42
Figura 4.21. Teste do algoritmo de content matching dos projetos.....	44
Figura 4.22. Teste do algoritmo de content matching das competências .....	46

## **LISTA DE ABREVIATURAS, ACRÓNIMOS E SIGLAS**

API	<i>Application Programming Interface</i>
BA	<i>Business Analysis</i>
CRISP-DM	<i>Cross-Industry Standard Process for Data Mining</i>
CS	<i>Cosine Similarity</i>
DB4S DB	<i>Database Browser for SQLite</i>
DL	<i>Deep Learning</i>
DS	<i>Desenvolvimento de Software</i>
FL	<i>Fuzzy Logic</i>
GP	<i>Gestão de Projetos</i>
HTML	<i>Markup Language</i>
IA	<i>Inteligência Artificial</i>
ITEMCF	<i>Item-based Collaborative Filtering</i>
JS	<i>JavaScript</i>
ML	<i>Machine Learning</i>
NN	<i>Neural Networks</i>
NLP	<i>Natural Language Processing</i>
PM	<i>Project Management</i>
PMBOK	<i>Project Management Body of Knowledge</i>
RSL	<i>Revisão Sistemática de Literatura</i>
SQL	<i>Structured Query Language</i>
TF-IDF	<i>Term Frequency-Inverse Document Frequency</i>

## INTRODUÇÃO

No âmbito do programa de Mestrado em Sistemas de Informação de Gestão, no Instituto Politécnico de Coimbra, foi proposto o tema “Composição de Equipas de Projetos de Desenvolvimento de *Software* com Inteligência Artificial”. Este estudo visa explorar como a Inteligência Artificial (IA) pode ser aplicada à Gestão de Projetos (GP), com foco na seleção dos colaboradores das equipas de projeto, com o objetivo de aumentar a eficiência e promover a inovação.

A IA tem como objetivo sistematizar e automatizar tarefas intelectuais, o que lhe dá um potencial relevante para diversas esferas de atividade, desde áreas de aplicação geral, como aprendizagem, perceção, as tarefas específicas como jogos de xadrez, demonstração de teoremas matemáticos, criação de poesia e diagnóstico de doenças (Gomes, 2010).

Vários estudos recentes mostram que a IA tem evoluído no quotidiano das empresas e dos consumidores como a PwC que vê as tecnologias de IA a evoluir ao longo dos anos fundamentalmente no cenário dos negócios. A grande maioria dos CEOs (91%) regionais acreditam que a IA mudará significativamente a forma como fazem negócios nos próximos cinco anos. No entanto, apenas 23% dos CEOs do Médio Oriente afirmam já ter introduzido iniciativas de IA, pouco mais de metade da percentagem global, o que sugere que as empresas da região precisam de recuperar rapidamente. Segundo os CEOs, existem duas coisas que estão a impedir as organizações regionais de avançar: dados e competências. Apenas 13% acreditam que a qualidade dos dados da sua empresa é adequada para avaliar as preferências e necessidades dos clientes. Cerca de 61% dos CEOs regionais citaram a falta de talento analítico como a principal razão para não terem informações adequadas (PwC, 2019).

Na era da Indústria 5.0 as empresas utilizam tecnologias avançadas, como a IA, para tornar a produção mais eficiente e alinhada com as necessidades humanas. De forma semelhante, no cenário empresarial contemporâneo, caracterizado por uma crescente complexidade e competitividade, a eficácia das equipas de projeto tornou-se um fator crucial para o sucesso organizacional. Assim, as empresas estão constantemente à procura

## *Composição de Equipas de Projetos de Desenvolvimento de Software com IA*

---

de métodos inovadores para melhorar a eficiência, reduzir custos e aumentar a produtividade das suas equipas, o que pode ser alcançado através dos benefícios da utilização da IA na GP (Taboada et al., 2023a).

Uma recente investigação do PMI (2019) confirma que a revolução da IA está em curso: 81% dos participantes indicam que a sua organização está a ser impactada por tecnologias de IA e 37% dos participantes referem que a integração da tecnologia é uma prioridade elevada para a organização que estimula a transformação nas abordagens de GP. Nos próximos três anos, os profissionais de GP antecipam que o número de projetos geridos com recursos à IA aumentem de 23% para 37%.

A IA com suas capacidades avançadas de análise de dados, aprendizagem automática e automação, oferece novas oportunidades para melhorar a dinâmica e o desempenho das equipas. A sua aplicação na GP pode não só aumentar a eficiência operacional, como também promover um ambiente de trabalho mais colaborativo e adaptável (Alshaikhi & Khayyat, 2021).

O tema “Composição de Equipas de Projetos de Desenvolvimento de *Software* com IA”, surge da necessidade de explorar como é que a IA pode ser integrada na GP, especialmente na composição das equipas de trabalho. Para isso, serão analisados casos das aplicações da IA na seleção de equipas competentes, atribuição de tarefas e algoritmos específicos que lidam com esta combinação.

Pretende-se também, melhorar os aspetos ligados com a eficiência e a qualidade da GP para criar equipas mais produtivas, capazes de enfrentar os desafios do ambiente atual de negócios e obter resultados mais sólidos e bem-sucedidos. Para isso serão analisadas as atividades desenvolvidas nos projetos de desenvolvimento de *software* e as competências necessárias dos membros da equipa para garantir o maior sucesso possível em projetos futuros com integração da IA. Serão analisadas as tecnologias atuais de IA, como *Machine Learning* ou Processamento de Linguagem Natural, bem como os seus impactos práticos no planeamento, execução e monitorização de projetos para proporcionar uma compreensão profunda das capacidades da IA na gestão de equipas de projeto.

## Composição de Equipas de Projetos de Desenvolvimento de Software com IA

---

No âmbito deste estudo proposto, os objetivos específicos foram delineados para direcionar a investigação de forma precisa e abrangente em função dos seguintes objetivos:

1. Análise de tipos projetos: Identificar os tipos de projetos de desenvolvimento de *software*.
2. Identificação de competências: Caracterizar as competências humanas necessárias para a execução de projetos de desenvolvimento de *software*.
3. Seleção de algoritmos: Realizar uma revisão de literatura da utilização da IA na composição das equipas de GP de desenvolvimento de *software*, com o objetivo de identificar os algoritmos mais utilizados.
4. Implementação e teste dos algoritmos selecionados: Implementar os algoritmos e testá-los em cenários práticos de GP de desenvolvimento de *software*.

A estrutura da dissertação é composta por seis capítulos, sendo este primeiro o da Introdução. De seguida apresenta-se o enquadramento teórico de suporte ao desenvolvimento do trabalho, com apresentação de alguns conceitos basilares como IA e GP. No terceiro capítulo apresenta-se a revisão de literatura realizada com o intuito de se descobrirem exemplos de utilização da IA na composição das equipas de desenvolvimento de projetos de *software*. O capítulo quarto apresenta a experiência realizada, que se baseia na utilização de um algoritmo de comparação de conteúdos (em inglês, *content matching*) que utiliza os mesmos princípios que os algoritmos de recomendação da filtragem colaborativa baseada em itens (em inglês, *Item-based collaborative filtering (ItemCF)*), uma técnica de recomendação popular utilizada para sugerir itens aos utilizadores com base na semelhança entre itens (Sarwar et al., 2001). Por fim, o capítulo da Conclusão, onde é feita uma breve síntese do trabalho desenvolvido e dos seus principais contributos, bem como as suas limitações e propostas de trabalhos futuros.

## 1 ENQUADRAMENTO TEÓRICO

Neste capítulo, apresentam-se os conceitos teóricos referentes à Inteligência Artificial (IA) e às tecnologias que incorporam esta nova competência das máquinas, propondo-a como uma solução inovadora para otimizar a gestão das equipas de projetos. A exploração desses conceitos visa proporcionar uma compreensão clara do potencial da IA para transformar práticas de gestão e qualidade sustentável dos projetos.

### 1.1 Inteligência Artificial

A IA é uma disciplina recente, que surgiu após a Segunda Guerra Mundial, com o passar dos anos tem evoluído exponencialmente e abrange uma vasta gama de avanços tecnológicos e inovações em diversos sectores, sendo a sua mais recente expressão o ChatGPT utilizado, por exemplo, na área da contabilidade (Dias et al., 2023) ou desenvolvimento de *software* (Colomo-Palacios et al., 2014).

Nos anos 40 e 50 começaram os primeiros esforços para criar máquinas que pudessem imitar a inteligência humana. Um dos marcos iniciais foi o modelo de neurónios artificiais proposto por Gomes (2010) que explorava os princípios da neurofisiologia para criar uma base teórica para redes neuronais artificiais. Durante os anos 60 e 70, a IA transformou em sistemas baseados em conhecimento, nos quais o conhecimento humano específico de domínio era codificado em programas de computador. Isso levou ao desenvolvimento de sistemas especializados capazes de realizar tarefas específicas, como diagnóstico médico e tradução de idiomas. A partir dos anos 80, a IA cresceu na indústria, surgiram sistemas comerciais e investimentos significativos das empresas e governos em investigação e desenvolvimento. Paralelamente, surgiram projetos ambiciosos, como o projeto Fifth Generation no Japão, para criar computadores baseados em IA através da linguagem de programação como o Prolog (Smith, 2006).

Em 2009, Alan Turing (2009) escreveu o artigo intitulado “*Computing Machinery and Intelligence*”, onde introduziu o famoso “Teste de Turing”, uma proposta para determinar se uma máquina exhibe comportamento inteligente que seja indistinguível do comportamento humano. Essencialmente, é uma experiência para avaliar a inteligência

## Composição de Equipas de Projetos de Desenvolvimento de Software com IA

---

das máquinas. A principal questão que Turing abordou em seu estudo foi: “Será que as máquinas conseguem pensar?”. Desde então, a IA passou por uma evolução notável.

Atualmente a IA tem impulsionado as indústrias com a capacidade de desenvolver sistemas inteligentes como as plataformas de *streaming* até assistentes virtuais como a Siri, desenvolvida pela Apple, e Alexa, desenvolvida pela Amazon, em dispositivos domésticos e móveis, que auxiliam nas tarefas como pesquisas na *web* e controlo de dispositivos domésticos (sigmoidal.ai, 2024). Além disso, sistemas de reconhecimento da fala, como o Google Voice, e de reconhecimento facial presentes em redes sociais e sistemas de segurança, são exemplos adicionais da utilização da IA. Carros autónomos, tradução automática (ex. Deepl ou ChatGPT) e diagnóstico médico são algumas das áreas onde a IA transforma os processos e serviços na sociedade (Pimenta et al., 2023; Smith, 2006).

A definição de IA varia consoante o contexto e a abordagem teórica adotada pelos investigadores. Gomes (2010) define-a como pensamento (relacionado com processos de pensamento e raciocínio) e ação (aborda o comportamento). A IA é considerada como a ciência de criar máquinas inteligentes, centradas no ser humano, que envolve observações e hipóteses sobre o comportamento humano, uma abordagem racionalista que combina matemática e engenharia. Essa definição foi explorada através de quatro perspetivas:

- Sistemas que pensam como seres humanos: procura criar máquinas capazes de pensar de forma semelhante aos seres humanos, tentando replicar o processo de pensamento humano.
- Sistemas que atuam como seres humanos: o foco principal, está na criação de máquinas que possam executar tarefas de forma inteligente, semelhantes às realizadas por seres humanos.
- Sistemas que pensam racionalmente: combina estudos da mente humana através do uso de modelos computacionais para compreender e replicar processos mentais.

*Composição de Equipas de Projetos de Desenvolvimento de Software com IA*

---

- Sistemas que atuam racionalmente: esta abordagem torna projeto de agentes inteligentes capazes de tomar decisões lógicas e racionais, independentemente de se assemelharem ao pensamento humano.

Neste sentido, é importante reconhecer que a definição da IA é um tema amplo considerado como uma parte da ciência da computação com o propósito de desenvolver máquinas inteligentes para benefício humano.

## **1.2 Gestão de Projetos**

Segundo António Miguel (2016), a GP é uma prática relativamente recente, caracterizada pela adaptação de técnicas especializadas de gestão, que visa proporcionar um melhor controlo e aproveitamento eficaz dos recursos disponíveis. Prática essa que está a ser implementada em diversas indústrias e organizações, desde empresas de produção e construção até setores farmacêuticos, hospitalares, contabilísticos e publicitários (Taboada et al., 2023).

Romero Izurieta et al. (2022), definem a GP como conjunto de tarefas e atividades realizadas para alcançar os objetivos esperados dentro de um prazo e recursos limitados. Além disso, consideram que para um projeto ser bem-sucedido, é importante considerar o sucesso do produto, a qualidade da solução, a satisfação das partes interessadas e impacto nos negócios, pois essas diferentes perspetivas estão inter-relacionadas e influenciam a coordenação das atividades do projeto, algo que é corroborado por outros autores na literatura (Varajão, Marques, et al., 2022; Varajão, Trigo, et al., 2022; Varajão & Trigo, 2016).

O PMI, a maior associação mundial de gestores de projeto, contribuem para esse campo através do PMBOK® (Guia para o Corpo de Conhecimento em GP) (PMI, 2017), um conjunto consolidado de conhecimentos e práticas adotadas em vários setores do mercado com interesse em GP. Além disso, o PMBOK descreve dez áreas de conhecimento da GP, pois fornece conceitos, ferramentas e técnicas específicas utilizadas nos grupos de processos que incluem: integração, âmbito, tempo, custo, qualidade, aquisições, riscos, recursos humanos, comunicação e partes interessadas Auth et al. (2021).

*Composição de Equipas de Projetos de Desenvolvimento de Software com IA*

---

O PMBOK descreve o ciclo de vida da GP como cinco fases, a saber (PMI, 2017):

1. **Iniciação:** Nesta fase, o projeto é oficialmente autorizado e os objetivos iniciais são definidos. Por outras palavras ocorre quando a equipa de projeto é selecionada e os recursos são alocados.
2. **Planeamento:** É definido o âmbito do projeto, as atividades são identificadas, os recursos são planeados e os cronogramas são desenvolvidos, com objetivo de criar um plano de projeto detalhado que orientará a execução e o controle do projeto.
3. **Execução:** O trabalho real do projeto é concretizado, sendo os recursos mobilizados, as atividades realizadas conforme planeado e as entregas produzidas de acordo com as especificações do projeto.
4. **Monitorização e Controlo:** Durante esta fase, o progresso do projeto é monitorizado e avaliado em relação ao plano. Os desvios são identificados e corrigidos conforme necessário para garantir que o projeto seja entregue conforme estipulado.
5. **Encerramento:** Esta fase marca o fim formal do projeto. Os resultados são entregues aos stakeholders, os recursos são libertados e as aprendizagens documentadas para futuros projetos.

### **1.3 Projetos de desenvolvimento de *software***

A Engenharia de *Software* é uma disciplina que envolve a produção de *software*, classificados por fases desde a iniciação de especificação de requisitos até a manutenção (ocorre quando o sistema está a ser utilizado). O desenvolvimento de *software* (DS) é um projeto em equipa que não se limita às tecnologias e metodologias, mas envolve aspetos humanos como as competências certas dos membros da equipa (Lima & Porto, 2019).

Liubchenko, (2022) define projetos DS como conjunto de atividades desenvolvidas para criar, projetar, desenvolver, testar e implementar *software*. Estas atividades são chamadas de etapas do processo de ciclo de vida do *software*, incluem etapas como a conceção, o planeamento, a manutenção e desinstalação do *software*.

### 1.3.1 Tipos de projetos

A GP de *software* envolve a utilização de competências, ferramentas e conhecimentos em engenharia de *software* para executar tarefas e atividades específicas, com o objetivo de responder aos requisitos e alcançar metas definidos pelo projeto. Isso inclui áreas como requisitos, design, desenvolvimento, teste, manutenção, suporte, gestão de configuração, metodologias de GP, processos de engenharia, modelação e simulação, gestão de qualidade, economia de engenharia e fundamentos de computação, matemática e engenharia (Radaideh, 2021).

Shafqat et al. (2021) abordam a existência de vários projetos de desenvolvimento de *software*, que podem variar consoante a natureza e os requisitos específicos dos tipos de projetos, esses são classificados por:

- Desenvolvimento inicial:
  - Projetos de desenvolvimento novo, em específico para o cliente
  - Projetos de desenvolvimento, de novos produtos de *software*
- Reutilização com modificação:
  - Projetos de atualização de novas versões de *software*
  - Projetos de desenvolvimento de tecnologia da informação e comunicação
- Projetos de desenvolvimento de integração de *software*
  - Reutilização sem modificação:
  - Projetos de configuração de *software* de pacote
  - Projetos de conversão de dados

### 1.3.2 Características dos colaboradores

As *soft skills* são importantes para engenheiros de *software*, competências como comunicação, liderança, resolução de conflitos e trabalho em equipe destacam a necessidade de colaboração, pensamento crítico, responsabilidade e ética (Lima & Porto, 2019).

As competências e experiência dos colaboradores nas organizações é fundamental para desenvolver produtos de alta qualidade e terminar os projetos com sucesso. A Tabela 1.1

*Composição de Equipas de Projetos de Desenvolvimento de Software com IA*

apresenta o conjunto de competências técnicas e não técnicas identificadas por Riaz et al. (2017), conjunto esse que se considerou suficiente para a realização do presente projeto.

Tabela 1.1. Competências dos colaboradores das equipas de desenvolvimento de software

<b>Competências técnicas</b>
Programação em diversas linguagens (Java, Python, C++, etc.).
Conhecimento em base de dados e modelagem de dados.
Desenvolvimento de interfaces de utilizador (UI/UX).
Testes de <i>software</i> e garantia de qualidade.
Arquitetura de <i>software</i> e design de sistemas.
<b>Competências não técnicas ou <i>soft skills</i></b>
Liderança e habilidades de gestão de equipe.
Comunicação verbal e escrita.
Resolução de problemas e tomada de decisão
Gestão de tempo e capacidade de organização.

Fonte: Adaptado de Riaz et al. (2017).

## 1.4 Inteligência Artificial e a Gestão de Projetos

A IA é considerada como uma ferramenta poderosa que auxilia em diversos setores da sociedade e proporciona uma série de benefícios significativos. Ao incorporar algoritmos avançados, a IA oferece uma série de benefícios e desafios para os gestores de projetos modernos e pode ser aplicada em inúmeros propósitos e áreas como a saúde, a segurança internacional, serviços bancários e financeiros.

Os resultados definidos por Taboada et al., (2023) demonstram um aumento significativo de publicações influentes sobre o uso da inteligência artificial nas últimas décadas, principalmente o *Machine Learning*, uma técnica aplicada na GP de construção e TI, com capacidades de prever riscos e na tomada de decisão, especialmente nos aspetos de planeamento, medição e tratamento de incertezas.

Segundo Alshaikhi & Khayyat (2021), a IA pode reduzir a mão-de-obra e as tarefas rotineiras podem ser substituídas por máquinas. A partir da perspectiva da GP, a IA pode imitar a mente humana na tomada de decisões e na resolução de problemas. Por outro

## *Composição de Equipas de Projetos de Desenvolvimento de Software com IA*

---

lado, existem métodos de IA que são utilizados em motores de busca e sistemas de reconhecimento de voz como Siri e Google Search.

Ao adotar e implementar sistemas de IA, as organizações podem ter benefícios como (Alshaikhi & Khayyat, 2021):

1. **Eficiência e Produtividade:** Automatiza as tarefas diárias, diminui a carga de trabalho dos recursos humanos para se concentrarem em atividades mais criativas e estratégicas.
2. **Tomada de Decisão Informada:** Analisa grandes volumes de dados para identificar padrões e tendências para fornecer *insights* valiosos na tomada de decisões informadas e estratégicas.
3. **Inovação e Competitividade:** Identifica oportunidades de melhoria e inovação para ajudar as organizações a se manterem competitivas em um ambiente de negócio.
4. **Qualidade dos Serviços:** Melhora a qualidade e a eficiência dos serviços, com informações mais precisas, recomendações personalizadas e suporte ao cliente mais eficaz.

No entanto, apesar desses benefícios, a adoção da IA também apresenta uma série de desafios e barreiras significativas devido (Alshaikhi & Khayyat, 2021):

1. **Compreensão e aceitação:** A falta de compreensão na adoção da IA sobre suas capacidades e benefícios potenciais, gera preocupações sobre o impacto no emprego e na sociedade em geral.
2. **Integração com sistemas existentes:** Integrar sistemas de IA com infraestruturas e processos existentes pode ser complexo e desafiador, exige investimentos adicionais em formação e desenvolvimento do pessoal.
3. **Regulamentações e normas:** Preocupações sobre a necessidade de regulamentações e normas para garantir seu uso ético e responsável, bem como proteger os direitos dos utilizadores e a privacidade dos dados.

*Composição de Equipas de Projetos de Desenvolvimento de Software com IA*

4. Desigualdades sociais e económicas: Existe o risco de que a adoção desigual da IA possa agravar as desigualdades sociais e económicas, pode gerar divisões entre aqueles que têm acesso à tecnologia e aqueles que não têm.

Ao enfrentar esses desafios de forma proativa e adotar uma abordagem estratégica para a integração da IA nos processos comerciais, as organizações podem aproveitar ao máximo o potencial das tecnologias com as ferramentas de automação em GP, como Chatbots, Zivebox, Rescoper, Clickup, Polydone e Clarizen, para aumentar a eficiência e a produtividade das equipas, facilitar a comunicação, atribuir e acompanhar o progresso das tarefas, a gestão de recursos e impulsionar a inovação no setor de atuação.

**1.4.1 Algoritmos de Inteligência Artificial**

À medida que existem avanços na compreensão da mente humana e na capacidade de criar sistemas computacionais sofisticados, a conceção de IA também pode mudar. A Tabela 1.2 apresenta as diversas técnicas que contribuem para o ecossistema da IA, conforme destacado por Taboada et al. (2023).

*Tabela 1.2. Algoritmo de IA na gestão de projetos*

<b>Algoritmo</b>	<b>Aplicabilidade</b>
<i>Machine Learning</i>	<ul style="list-style-type: none"> <li>• Previsão de prazos e custos</li> <li>• Acompanhamento de projetos</li> <li>• Gestão do risco</li> <li>• <i>Alocação/Gestão</i> de recursos (incluindo gestão de pessoas)</li> <li>• Tomada de decisão</li> </ul>
<i>Fuzzy logic</i>	<ul style="list-style-type: none"> <li>• Avaliação de riscos</li> <li>• Estimativa de prazos</li> <li>• Estimativa de custos</li> <li>• Tomada de decisão</li> <li>• Atribuição de recursos</li> <li>• Monitorização do desempenho</li> <li>• Gestão da qualidade</li> </ul>
<i>AI-based heuristics (Genetic Algorithms (GAs))</i>	<ul style="list-style-type: none"> <li>• Otimização do calendário das tarefas</li> <li>• <i>Alocação/Gestão</i> de recursos</li> <li>• Gestão do risco</li> </ul>

*Composição de Equipas de Projetos de Desenvolvimento de Software com IA*

<i>and Ant Colony Optimization (ACO)</i>	<ul style="list-style-type: none"> <li>• Tomada de decisão</li> </ul>
<i>Natural Language Processing</i>	<ul style="list-style-type: none"> <li>• Extração de informação (identificação de pontos críticos e tendências que podem influenciar os resultados do projeto).</li> <li>• Gestão da comunicação</li> <li>• Identificação e gestão de riscos</li> <li>• Análise de requisitos</li> <li>• Automatização de tarefas de rotina</li> <li>• Análise de sentimentos (membros da equipa e das partes interessadas)</li> </ul>
<i>Neural Networks</i>	<ul style="list-style-type: none"> <li>• <i>Previsão de resultados</i></li> <li>• <i>Seleção de projetos e gestão de carteiras</i></li> <li>• <i>Estimativa de custos e prazos</i></li> <li>• <i>Gestão de riscos</i></li> <li>• <i>Alocação/Gestão de recursos</i></li> <li>• <i>Previsão de desempenho</i></li> </ul>
<i>Deep Learning</i>	<ul style="list-style-type: none"> <li>• Previsão de custos</li> <li>• Avaliação da atividade da mão de obra</li> <li>• Segurança no local de construção</li> <li>• Monitorização e previsão do estado de saúde estrutural</li> <li>• Programação de projetos e previsão de duração</li> </ul>

Fonte: Adaptado de Taboada et al. (2023)

Essas técnicas abordam conceitos desde otimização de recursos até alocação de tarefas à tomada de decisão, demonstrando a diversidade de práticas de IA utilizadas na GP. Em resumo, Taboada et al. (2023) refere que os algoritmos como *Machine Learning* (ML), *Fuzzy Logic* (FL), *Neural Networks* (NN) e *Natural Language Processing* (NLP) podem apoiar a GP, enquanto que os algoritmos heurísticos baseados em IA podem melhorar o desempenho da equipa.

#### **1.4.2 Processamento de Linguagem Natural**

Um dos algoritmos escolhidos para o presente projeto é o Processamento de Linguagem Natural (NLP, do inglês *Natural Language Processing*). O NLP surgiu nos anos 1950, caracterizado como uma área da IA focada na interação entre computadores e linguagem humana que utiliza técnicas estatísticas para indexar e pesquisar grandes volumes de texto de forma eficiente. Ao combinar técnicas de linguística, ML e processamento de texto, o

*Composição de Equipas de Projetos de Desenvolvimento de Software com IA*

---

NLP capacita os computadores a compreender, interpretar e até mesmo gerar linguagem humana de maneira semelhante aos seres humanos (Niranjan et al., 2016)

Neste sentido, uma das aplicações mais proeminentes do NLP é a análise de sentimentos em redes sociais, onde algoritmos são utilizados para identificar opiniões e emoções expressas em *posts*, *tweets* e comentários. É vantajoso para as empresas porque os ajuda a entender o *feedback* do cliente, mas também a moldar estratégias de marketing e tomada de decisão. Por outro lado, NLP desempenha um papel vital na tradução automática, permitindo que os computadores convertam texto de um idioma para outro com precisão e fluência crescentes. Isso é especialmente útil em um mundo globalizado, onde a comunicação entre culturas e idiomas é essencial para o comércio internacional, cooperação científica e compreensão intercultural (Nadkarni et al., 2011).

Outra aplicação importante do NLP é a extração de informações de grandes volumes de texto, como documentos legais, artigos de jornais e registos médicos. Os algoritmos de NLP podem identificar entidades nomeadas, como nomes de pessoas, locais e datas, além de relacionamentos entre elas para obter uma análise rápida e eficiente de grandes conjuntos de dados.

## 2 REVISÃO DE LITERATURA

Felipa Reis (2018), define a Revisão Sistemática de Literatura (RSL), como uma metodologia que permite levantar dados bibliográficos de trabalhos para aprofundar o conhecimento teórico na área de estudo e avaliar os contributos dos autores sobre o tema em análise. Neste sentido, a RSL tem como objetivo proporcionar aos leitores uma visão abrangente do estado atual de todas as fontes de pesquisa existentes sobre o tema de investigação “Otimização das Equipas de Projeto com Inteligência Artificial”, soluções para melhorar a eficiência, produtividade e qualidade dos projetos, bem como estabelecer uma base sólida para estudos futuros através das técnicas de IA.

Kitchenham (2004) apresenta a abordagem sobre a RSL como um processo que envolve a identificação, avaliação e interpretação de todos os intervenientes relevantes disponíveis durante uma pesquisa específica. Esta abordagem destaca três pilares fundamentais: o planeamento, a execução e a divulgação da investigação, os quais são classificados em fases distintas, tais como:

- Identificação da necessidade de revisão sistemática;
- Seleção dos estudos primários;
- Extração e monitorização de dados;
- Integração de dados e divulgação.

### 2.1 Planeamento

Após o enquadramento da RSL, a fase inicial desta investigação utilizou a seguinte expressão de pesquisa, para encontrar estudos para construção de uma rede bibliométrica que pudesse ajudar na descoberta de novas palavras chave:

***ALL ("software" AND "project management" AND ("human resource" OR "human allocation") AND ("artificial intelligence" ) )***

A inclusão de filtros foi fundamental, para restringir a busca em artigos específicos das publicações no intervalo de período entre 2019 a 2023. Esses critérios foram selecionados para direcionar os resultados conforme o foco desejado e reduzir a quantidade de dados a

*Composição de Equipas de Projetos de Desenvolvimento de Software com IA*

serem considerados para a pesquisa. Os resultados foram unificados com a eliminação de artigos duplicados e com filtragem de acordo com critérios de seleção.

A base de dados utilizada foi a SCOPUS, porque indexa publicações de alta qualidade com revisão por pares, além de permitir a exportação de dados da pesquisa em vários formatos. As publicações obtidas foram analisadas com a ferramenta Vosviewer (2024), um *software* que cria e permite visualizar redes bibliométricas<sup>1</sup>, tendo resultado dessa análise a rede bibliométrica apresentada na Figura 2.1.

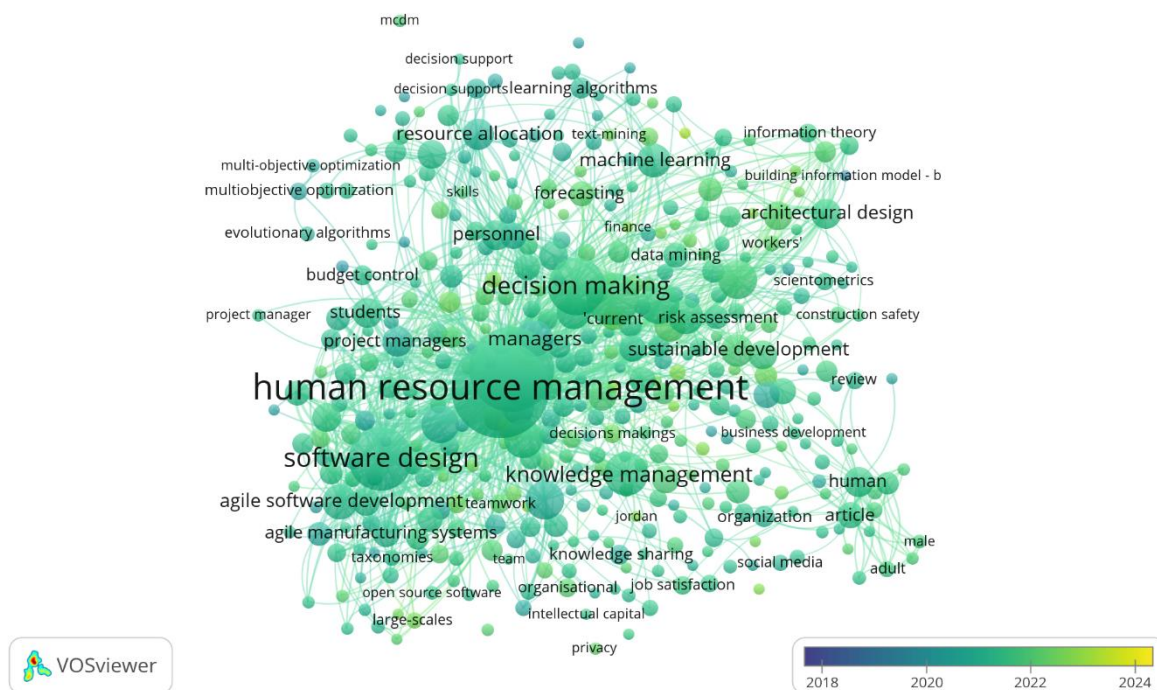


Figura 2.1. Rede bibliométrica obtida a partir da primeira expressão de pesquisa

Após a análise da rede bibliométrica resultante desta primeira pesquisa descobriram-se algumas palavra-chave como “*Human resource management*”, “*Personnel*”, “*Skills*”, “*Team*”, “*Machine Learning*” e “*Data Mining*”, bem como palavras relacionadas com a área de otimização e algoritmos de genéticos, mas que não irão ser consideradas na

<sup>1</sup> As redes bibliométricas incluem atores ou publicações individuais e podem ser construídas com base em citações, ligação bibliográfica, cocitação para identificar e examinar as conexões e padrões emergentes com objetivo de compreender a estrutura e as interações subjacentes de forma a fornecer insights valiosos para futuras investigações e intervenções.

## *Composição de Equipas de Projetos de Desenvolvimento de Software com IA*

---

construção da segunda expressão de pesquisa por não estarem diretamente relacionadas com o tema da IA.

Após este estudo inicial sobre o tema em estudo, definiu-se o protocolo para a condução da RSL que incluiu, entre outros, a definição da expressão de pesquisa e das bases de dados a utilizar para a RSL. A expressão de pesquisa utilizada foi a seguinte:

```
TITLE-ABS-KEY ("software project management" AND ("human resource" OR "human allocation" OR "personnel" OR "team*" OR "skill*" OR "compet*")) AND ("artificial intelligence" OR "machine learning") ) AND PUBYEAR > 2018 AND PUBYEAR < 2024 AND ( LIMIT-TO ( DOCTYPE , "cp" ) OR LIMIT-TO ( DOCTYPE , "ar" ) )
```

*Figura 2.2. Expressão de pesquisa da RSL*

Não obstante, se poderem escolher várias bases de dados para as pesquisas, optou-se pela base de dados SCOPUS, à semelhança do que tinha sido feito na primeira pesquisa, uma base de dados que se destaca por ser a única base de dados que combina dados automatizados e manual para criar perfis de autores atuais com mais de 40 idiomas. Este processo disponibiliza mais de 17 milhões de perfis e facilita a pesquisa em concreto de autores e artigos.

## **2.2 Execução**

Nesta etapa, segundo Kitchenham (2004), são executadas as seguintes etapas: identificar os estudos; selecionar os estudos; avaliar a qualidade; extrair os estudos; e apresentar uma síntese dos estudos. De seguida, apresentam-se as etapas relativas a execução da RSL segundo Kitchenham (2004).

### **2.2.1 Identificar os estudos**

Na fase inicial da RSL, foi definido o processo de identificação e seleção rigorosa dos estudos relevantes. Para tal, utilizou-se a base de dados SCOPUS e a expressão de pesquisa apresentada na Figura 2.2, o que resultou na recolha de 10 documentos. A identificação dos registos foi realizada com base no tipo de documento, classificado por artigos e documentos de conferência. Além disso, foram selecionados apenas artigos em inglês, uma vez que este é amplamente reconhecido como o idioma predominante na comunicação científica e académica devido a maior diversidade de fontes e pesquisas.

### **2.2.2 Selecionar os estudos**

Nesta etapa definem-se os critérios de elegibilidade dos estudos que podem ser critérios de inclusão ou de exclusão. O critério de elegibilidade foi estabelecido para guiar o processo de seleção dos estudos que serão incorporados na pesquisa para garantir apenas a inclusão de contribuições relevantes e robustas, a fim de garantir a qualidade da investigação. Nesse sentido, foram considerados estudos empíricos e revisões sistemáticas publicados nos últimos 7 anos (2018-2024), restritos às atas de conferências e revistas científicas que apresentassem casos práticos na aplicação da IA na GP. Outro critério de inclusão utilizado foram exemplos de casos de estudo que explorassem conceitos como: algoritmos, conjuntos de dados, modelos ou *frameworks* específicos de IA voltados para a otimização de tarefas em equipas de projeto referentes a área Engenharia de *Software*, seleção de artigos em inglês por ser o idioma predominante nas publicações.

Foram excluídos artigos que não abordassem diretamente a aplicação da IA na gestão de equipas de projetos ou que não fossem focados na área de desenvolvimento de *software*. Também foram excluídos estudos sem dados mensuráveis ou resultados pouco claros. Exclusão de títulos relacionados a pesquisas secundárias, revisões narrativas e sistemáticas, bem como artigos cujo conteúdo não foi possível obter, por ser necessário pagar pelo mesmo. Após aplicados estes critérios foram selecionados 10 dos 12 documentos iniciais.

### **2.2.3 Qualidade**

Este tópico já está contemplado na expressão de pesquisa. No entanto, como é uma das etapas da RSL de acordo segundo Kitchenham (2004) refere-se novamente que só foram considerados para esta RSL estudos de conferências ou revistas científicas.

### **2.2.4 Extrair os estudos**

No final, após a realização da RSL, foram extraídos 10 textos completos. Após a leitura, todos foram considerados.

*Composição de Equipas de Projetos de Desenvolvimento de Software com IA*

## 2.2.5 Síntese dos estudos

A Tabela 2.1 apresenta a lista dos artigos analisados, contendo informações sobre o autor, título do artigo técnicas e métodos analisados.

Tabela 2.1. Lista de artigos resultante da RSL

<b>Autores</b>	<b>Título</b>	<b>Técnicas/Métodos</b>
Kamthan & Shahmir (2019)	On <i>software</i> projects in academia and industry from a perspective of <i>software</i> engineering education	Inclusão de práticas e ferramentas utilizadas na indústria nos cursos,
Nethravathi et al. (2020)	A Redesigning <i>Software</i> Procedure in Improved <i>Software</i> Management using Machine Learning	Integração do ML no SDLC, desde a recolha de dados até ao treino e implantação dos modelos.
Ma et al. (2020)	A data-driven risk measurement model of <i>software</i> developer turnover	Utiliza o algoritmo K-means para agrupar eventos em diferentes atividades de projeto e calcular o risco de rotatividade nas atividades.
Mamatha & Suma (2021)	Role of machine learning in <i>software</i> project management	Desenvolvimento e treino do modelo Machine Learning para automatizar a atribuição das tarefas de equipas.
Nitin Rajadhyaksha & Saini (2022)	Robotic Process Automation for <i>Software</i> Project Management	Aplicação de RPA em diferentes fases do ciclo de vida do desenvolvimento de <i>software</i> .
Venkata Ramana & Narsimha (2022)	Identification of the Ideal Team Capabilities and Predictive Success Measure for <i>Software</i> Projects Using Machine Learning	Modelo de regressão múltipla desenvolvido para prever as condições dos projetos.
Mushtaq et al. (2022)	Early Size and Effort Estimation of Mobile Application Development	Combina o Modelo de Estimativa de Esforço COSMIC Plus com técnicas de Machine Learning.
Kanbur et al. (2023)	Creative AI in <i>Software</i> Project Management	Utiliza a IA para melhorar a precisão nas estimativas e automatizar as tarefas administrativas.
Younisse & Azzeh (2023)	Application of Natural Language Processing Techniques in Agile <i>Software</i> Project Management: A Survey	Analisa modelos de NLP, como Redes Neurais Recorrentes (RNNs) e Redes Neurais Convolucionais (CNNs),
Ahmadi et al. (2023)	Learning <i>Software</i> Project Management From Analyzing Q&A's in the Stack Exchange	Técnicas de Machine Learning como Random Forest, Support Vector Machine (SVM) e Naive Bayes.

## 2.3 Comunicação

A etapa da comunicação (em inglês, *Reporting*) é a etapa onde entre outros aspetos são apresentados e discutidos os resultados (Kitchenham, 2004), onde se apresentam as contribuições relevantes, métodos eficazes de alocação e a exploração de fatores determinantes para o sucesso das equipas de desenvolvimento de *software*, com o objetivo de proporcionar uma visão abrangente e fundamentada sobre o estado atual da literatura acerca da otimização das equipas de projeto com recurso à IA.

Felipa Reis (2018) refere-se a análise e discussão dos resultados como um dos pilares importantes da pesquisa científica, onde as descobertas são examinadas em profundidade e exploradas na literatura existente. O objetivo consiste em interpretar os dados obtidos, explorar suas implicações, relevância e possíveis limitações. Neste contexto, a análise dos artigos científicos permitiu identificar as técnicas de otimização de equipas até à análise de algoritmos e à síntese dos resultados mais relevantes.

Foram combinados os resultados obtidos e o estado atual do conhecimento sobre o tema em estudo, nomeadamente com o objetivo de:

- Aumentar a compreensão e permitir identificar os materiais existentes na RSL;
- Categorizar e examinar os dados para atender os objetivos da investigação;
- Organizar e comparar os dados para dar resposta à questão da investigação e a viabilidade ou não dos seus pressupostos.

Após a recolha dos documentos, a próxima etapa foi a descrição dos resultados. Nesta etapa foram analisados os temas de investigação relacionados com a formação das equipas de desenvolvimento de *software*, incluindo técnicas e algoritmos de alocação de recursos humanos.

A análise apresentada na Figura 2.3 revela a diversidade das áreas de atuação, destacando-se a área da Ciência da Computação (27%) e da Engenharia (26%) com as maiores percentagens.

*Composição de Equipas de Projetos de Desenvolvimento de Software com IA*

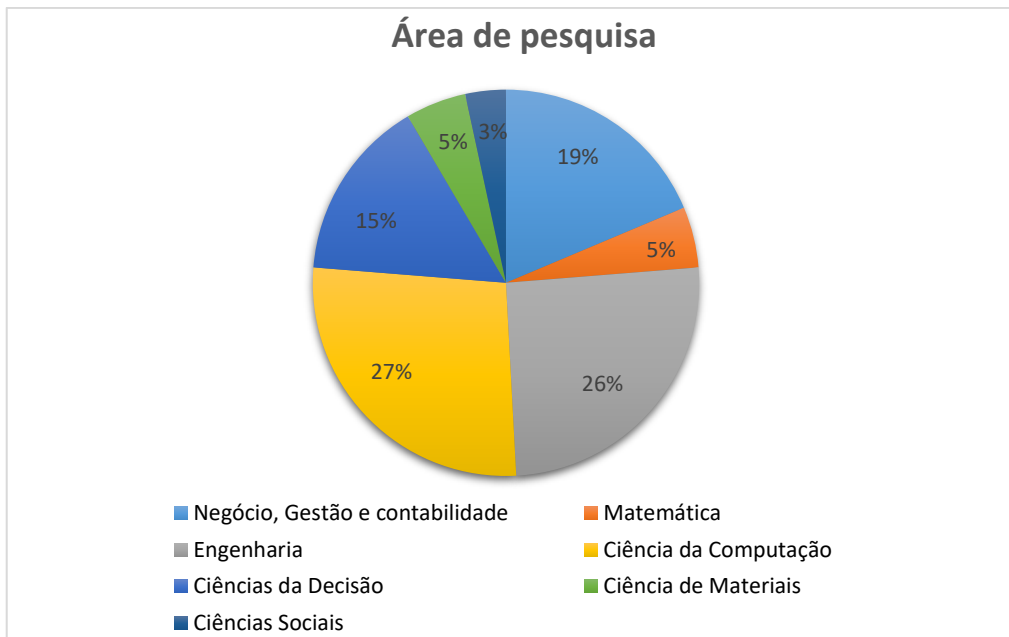


Figura 2.3. Área de pesquisa

Quanto à análise dos números de documentos por país, foi possível visualizar que a maior parte dos artigos foram publicados por autores de países diferentes como: Índia (7 artigos), China (4 artigos), Chipre (3 artigos), Estados Unidos (2 artigos) e Canadá (1 artigo).

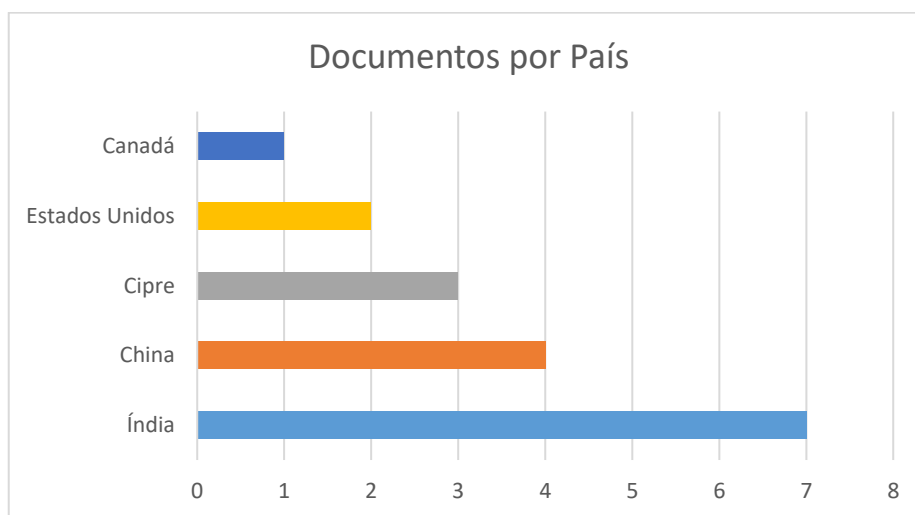


Figura 2.4. Número de documentos por país

Nos parágrafos seguintes apresentam-se os maiores contributos dos artigos selecionados, apresentados anteriormente na Tabela 2.1.

## Composição de Equipas de Projetos de Desenvolvimento de Software com IA

---

As organizações adotam a integração da IA na GP de *software* para automatizar as tarefas. Neste contexto, Kanbur et al. (2023) demonstra alguns benefícios da IA na melhoria da precisão da estimativa do tempo, automatização das tarefas administrativas, previsão dos riscos em função dos dados históricos, redução dos custos, simplificação do planeamento dos projetos e obtenção de maior eficiência e eficácia na GP o que resulta numa entrega mais rápida. Por outro lado, existe uma possível desvantagem, a dependência excessiva de algoritmos e automação, o que pode levar a decisões erradas se os dados de entrada estiverem incorretos. Outra desvantagem é a substituição das competências humanas e perda de aspetos qualitativos na gestão dos projetos, bem como a intuição e a experiência prática.

Preocupado em prever o sucesso ou fracasso da GP de *software*, os autores Venkata Ramana & Narsimha (2022) propõem duas soluções, o modelo de regressão múltipla, uma técnica estatística utilizada para entender a relação entre uma variável dependente e duas ou mais variáveis independentes, o resultado demonstrou uma taxa de acerto de 98,3% dos projetos concluídos com sucesso. E o modelo de sistema de recomendação com objetivo de formar equipas qualificadas para os projetos de varias dimensões, o resultado foi positivo com uma taxa alta de números de projetos concluídos com sucesso. Para isso, foram utilizadas três variáveis classificadas como idade, experiência e competências, combinados com os algoritmos de *Machine Learning* (ML) para analisar dados como a faixa etária, histórico de trabalhos e competência dos membros da equipa para criar combinações equilibrada de diversas áreas. Isso permitiu gerir os projetos de *software* para melhorar a eficiência na resolução de problemas complexos, diminuir os riscos de falha, garantir uma melhor adaptação às necessidades e expectativas dos clientes, contribuindo assim para o sucesso global dos projetos.

Kamthan & Shahmir, (2019) destacam as lacunas enfrentadas nos projetos académicos e industriais de engenharia de *software*, ou seja, comparam os projetos realizados nas universidades com os da indústria. Demonstram que os projetos académicos são muitas vezes mais simples e não refletem os desafios reais que os engenheiros de *software* enfrentam no trabalho. Para resolver isso, o artigo sugere que os professores devem incluir práticas e ferramentas utilizadas na indústria nos cursos e colaborar com empresas para

*Composição de Equipas de Projetos de Desenvolvimento de Software com IA*

---

dar aos estudantes uma experiência mais prática através de projetos reais, estágios, mentorias e *hackathons* para estarem preparados para o mercado de trabalho e investirem em formação com as exigências atuais da indústria para os seus futuros empregos na área de *software*.

Rajadhyaksha & Saini (2022) complementam a abordagem anterior, destacam a importância de alinhar a formação académica com as do mercado, neste caso os autores, implementa a tecnologia de Automação de Processos Robóticos (RPA), uma área em que a eficiência e a precisão são cruciais para automatizar as tarefas repetidas baseadas em regras, realizadas por humanos em sistemas digitais. Os resultados apresentados mostram como a RPA pode ser aplicada em diferentes fases do ciclo de vida do desenvolvimento de *software*, como a análise de requisitos, que ajuda na recolha e organização de dados dos *stakeholders*, acelera o processo na tomada de decisão e garante que todos os requisitos sejam cumpridos. O desenvolvimento, automatiza as tarefas como teste de unidade e integração para reduzir erros humanos e melhora a qualidade do código. Por fim, na instalação e manutenção que facilita na execução de testes de aceitação, gerir as atualizações de *software* e monitorizar a integridade do sistema. Isso resulta numa entrega mais rápida de *software* de alta qualidade, redução de erros e melhor utilização dos recursos disponíveis.

Nethravathi et al., (2020) abordam o papel do ML na engenharia de *software*, uma ferramenta essencial para analisar dados e extrair insights que permitem identificar padrões para melhorar continuamente os processos organizacionais, na gestão da mudança e no aumento da qualidade e produtividade dos produtos de *software*. Destacam a integração do ML no ciclo de vida do desenvolvimento de *software* (SDLC, do inglês *Software Development Life Cycle*), desde a recolha de dados até ao treino do modelo e sua implantação de forma a aumentar o foco conceitual do desenvolvimento. Os autores também destacam as diferentes abordagens do algoritmo de ML: o supervisionado, onde os dados são rotulados e o algoritmo aprende a associá-los a respostas corretas; o não supervisionado, permite que o algoritmo descubra padrões nos dados sem rótulos; e o semi-supervisionado, que utiliza uma combinação de dados rotulados e não rotulados. Essas abordagens oferecem flexibilidade aos cientistas de dados para escolher a melhor

*Composição de Equipas de Projetos de Desenvolvimento de Software com IA*

---

estratégia de acordo com a natureza dos dados e os objetivos do projeto em várias áreas de aplicação como, previsão de carga de energia, negociação de ações e diagnóstico médico.

Ambos os artigos utilizam as técnicas de ML na GP de *software*. Enquanto Nethravathi et al. (2020) se focam na análise de dados de comunidades para melhorar a gestão, Mamatha & Suma, (2021) propõe um algoritmo de ML para alocação eficiente de recursos humanos. Essa abordagem envolve o desenvolvimento e treino de um modelo de ML para automatizar a atribuição de tarefas aos membros da equipe através de dados históricos de projetos e identificação de padrões que possam auxiliar na tomada de decisões sobre quem deve realizar cada tarefa, de acordo com as habilidades e competências necessárias para determinadas atividades e atribuir essas tarefas aos membros da equipe com as habilidades mais adequadas. A solução implementada, permitiu otimizar a alocação de recursos humanos, melhorar a eficiência e a produtividade do projeto, possibilitou aos gerentes de projeto economizarem tempo na alocação manual de recursos e a se concentrarem em atividades mais estratégicas.

Por outro lado, Ma et al. (2020) propõem um modelo para medir o risco da rotatividade dos programadores em projetos de *software*, através de algoritmos para analisar registos de eventos e atribuir importância aos programadores. Este método inclui a utilização do algoritmo *K-means* para agrupar os eventos em diferentes atividades de projeto e permite calcular o risco associado à saída ou substituição de programadores em cada atividade. Por outro lado, é aplicado um algoritmo baseado em entropia da informação para medir este risco individualmente em cada atividade e, posteriormente, combiná-los para obter o risco global do projeto. Os resultados fornecem uma visão detalhada dos fatores que influenciam a rotatividade de programadores e permitem tomar medidas para evitar riscos e otimizar a GP de *software*.

Mushtaq et al. (2022), também destacam aspetos relacionados com a estimativa de esforço em projetos de *software*, para garantir que os projetos sejam concluídos dentro do prazo e do orçamento estabelecidos ele destaca a importância de estimar corretamente os esforços necessários no desenvolvimento numa aplicação movel. Para isso, propõem uma metodologia que combina o Modelo de Estimativa de Esforço COSMIC Plus

*Composição de Equipas de Projetos de Desenvolvimento de Software com IA*

---

(CPEEM) com técnicas de ML. Esta abordagem visa calcular os esforços com base em parâmetros específicos de desenvolvimento de aplicativos móveis, obtendo um resultado mais preciso. Os resultados obtidos são validados estatisticamente e comparados com os esforços reais de desenvolvimento. A pesquisa contribui para melhorar a precisão das estimativas de custo para o desenvolvimento de aplicações móveis, reduzir a diferença entre os esforços estimados e real.

Ahmadi et al. (2023) utilizam técnicas de ML para analisar dados de comunidades online de GP. Para isso, foram analisados um conjunto de dados com 5000 perguntas feitas na comunidade de GP no Stack Exchange ao longo de dez anos. Através da utilização das técnicas de ML e incorporação de texto, como BERT (Bidirectional Encoder Representations from Transformers) e Doc2Vec, foram comparados três algoritmos de ML para classificar e analisar os dados: o *Random Forest* cria uma “floresta” de árvores de decisão para classificação e regressão, o *Support Vector Machine (SVM)* encontra o hiperplano que melhor separa as classes em um espaço de dimensão superior e o *Naive Bayes* utiliza o teorema de Bayes para assumir independência entre os recursos. A combinação de *BERT* com os algoritmos *Random Forest* e *SVM* permitiu obter os melhores resultados para a gestão de cronograma como a área mais questionada e o planeamento como o grupo de processos mais abordado.

Por fim, Younis & Azzeh (2023) utilizam o NLP para melhorar a precisão das estimativas de esforço em metodologias ágeis, com base no conteúdo contextual das histórias de utilizadores. O estudo revelava uma tendência crescente na utilização de modelos de NLP, como Redes Neurais Recorrentes (RNNs) e Redes Neurais Convolucional (CNNs), para prever com precisão os pontos de história e o esforço necessário para concluir os projetos. Contudo, identificaram algumas limitações, como a escassez de conjuntos de dados específicos e a necessidade de desenvolver abordagens mais robustas para melhorar a eficácia dos modelos de NLP na estimativa de esforços.

Esses estudos revelam uma tendência emergente na aplicação de técnicas avançadas da inteligência artificial para enfrentar os desafios na engenharia de *software* e GP.

### 3 METODOLOGIA

A utilização de metodologias apropriadas é essencial para reduzir riscos em projetos de desenvolvimento, especialmente em projetos de ciência de dados, nos quais surgiram várias abordagens como modelos de processos de conhecimento e mineração de dados (KDDM, do inglês *Knowledge Discovery & Data Mining*). Estes modelos padronizam as etapas de identificação, qualificação, extração, tratamento, modelação e avaliação de dados.

Segundo Fernandes (2019) e Laureano et al. (2014) o modelo *Cross – Industry Standard Process for Data Mining* (CRISP-DM) é um dos modelos mais populares para a realização de projetos de ciência de dados, eleito por 42% dos profissionais (Laureano et al., 2014). O modelo CRISP-DM caracteriza-se por possuir um conjunto de técnicas e táticas que para orientar o processo de realização de projetos de ciência de dados, podendo ser representado por uma esquema circular composto por seis fases (Figura 3.1) de acordo a necessidades de cada utilizador, que são (Clinton et al., 1999): entendimento do negócio; entendimento dos dados; preparação dos dados; modelação; avaliação e entrega.

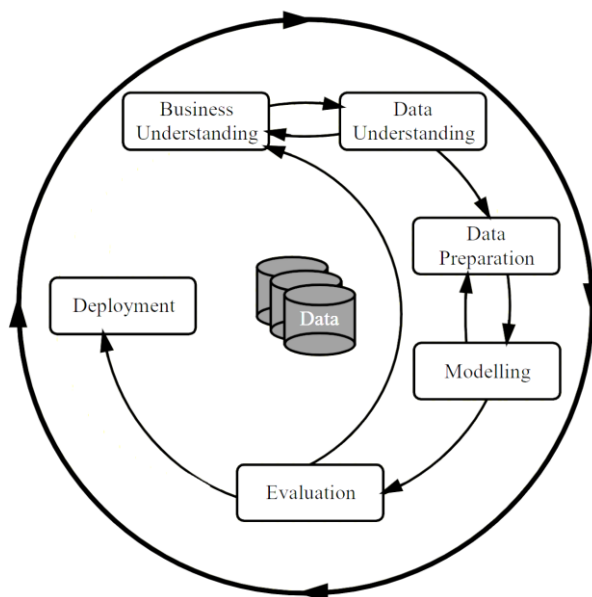


Figura 3.1. Modelo CRISP-DM

Fonte: Wirth & Hipp (2000)

*Composição de Equipas de Projetos de Desenvolvimento de Software com IA*

---

A Figura 3.1 apresenta o modelo CRISP-DM, cuja primeira fase é a compreensão dos objetivos e requisitos empresariais. Segue-se a recolha inicial dos dados para identificar problemas de qualidade e subconjuntos relevantes, preparando-os através da seleção, transformação e limpeza necessárias para a modelação. Na fase de modelação, diversas técnicas são aplicadas com ajustes de parâmetros para otimização. Após construir o modelo, é crucial avaliá-lo para garantir que corresponda aos objetivos definidos, identificando eventuais ajustes necessários. Finalmente, a entrega não se limita à disponibilização do modelo em produção, envolve também a organização dos resultados de forma acessível aos clientes, que incluem os relatórios e implementação de processos contínuos de DM.

Assim, a estrutura do capítulo da experiência realizada do relatório, que a seguir se apresenta, segue esta estrutura, possuindo os subcapítulos os nomes das diferentes fases do modelo, desde a análise detalhada do negócio para atender às necessidades dos gestores até à entrega da solução, entrega essa que não passou da fase de protótipo, no momento da realização do relatório esperando-se, no entanto, que uma solução melhorada possa vir a integrar e suportar a GP de *software* da empresa em questão.

## 4 EXPERIÊNCIA REALIZADA

Este capítulo apresenta as fases do trabalho de composição das equipas de projeto de desenvolvimento de *software* com recurso à IA de acordo com as fases da metodologia CRISP-DM. Essas fases englobam o entendimento do negócio, entendimento dos dados, preparação dos dados, modelação, avaliação e entrega da solução final.

### 4.1 Entendimento do negócio

A afetação de recursos humanos em projetos de desenvolvimento de *software* envolve a atribuição das pessoas certas às tarefas certas para garantir que os projetos são concluídos de forma eficiente e eficaz. Este processo é multifacetado e requer um conhecimento profundo das competências, experiência e carga de trabalho de cada membro da equipa. Os passos normalmente envolvidos na afetação de recursos humanos são a definição dos requisitos do projeto; enumerar as competências e experiência; atribuir funções e responsabilidades; e monitorizar e fazer ajustamentos às equipas se necessário. O presente trabalho foca-se nas etapas iniciais da afetação de recursos humanos que é a de tentar encontrar as pessoas certas para a realização das tarefas dos projetos com base na descrição do projeto. Para tal, criaram-se dois algoritmos de *content matching* utilizando técnicas de NLP e de ItemCF: o primeiro algoritmo procura identificar com base na descrição do projeto quais os melhores profissionais para a execução do mesmo; e o segundo algoritmo procura por projetos similares através da descrição dos mesmos, de forma a que o gestor de projetos possa avaliar qual a equipa de profissionais mais adequada ao projeto, que terá por base a similaridade e o Índice de Sucesso dos Projetos (ISP), um valor em percentagem de 0 a 100% que pretende transmitir o grau de sucesso do projeto.

### 4.2 Entendimento dos dados

Os modelos de dados apresentados são estruturados para atender às necessidades de um sistema de informação de gestão de projetos que possibilite a composição de equipas de desenvolvimento de *software* utilizando algoritmos de *content matching*. A escolha dos dados é baseada na natureza das informações que são cruciais para a análise e tomada de

## *Composição de Equipas de Projetos de Desenvolvimento de Software com IA*

---

decisão na formação de equipas e alocação de recursos, assim, como para a modelação e teste do modelo subjacente ao trabalho definiram-se os seguintes conjuntos de dados com as seguintes variáveis:

- **Conjunto com Projetos (ver apêndice 1):** os atributos dos projetos incluem nome, descrição, data de início, data de fim, equipas que participaram e o ISP. Esses dados permitem entender os requisitos específicos de cada projeto e o sucesso na execução do mesmo.
- **Conjunto com Colaboradores (ver apêndice 2):** os atributos dos colaboradores incluem as suas competências e experiências anteriores em projetos relevantes. Esses atributos são fundamentais para avaliar as competências de cada colaborador e se estão alinhadas com os requisitos do projeto em avaliação.
- **Conjunto com Equipas (ver Figura 4.3):** os atributos da equipa incluem somente o nome e a descrição da mesma.

Este projeto pretende ser uma prova de conceito (em inglês, *proof of concept*) para testar a exequibilidade da ideia. Em fases posteriores do projeto os conjuntos de dados poderão ser estendidos para incluir outras características que se considerem importantes, bem como, serem adicionados novos conjuntos de dados como, por exemplo, um conjunto com as tarefas realizadas por cada colaborador.

### **4.3 Preparação dos dados**

Neste tópico adotou-se a metodologia definida pelo Niranjana et al. (2016) que apresenta uma abordagem baseada nas etapas de pré-processamento e extração dos dados. A preparação dos dados é uma etapa crucial que envolve identificar fontes de dados relevantes, recolher, organizar e limpar os dados para garantir a sua consistência e integrá-los de forma adequada. Para este trabalho foram gerados dados sintéticos no ChatGPT para criar os conjuntos de dados referidos na secção anterior (ver apêndice 1 e 2).

#### **4.3.1 Geração de dados sintéticos**

Esta secção descreve o processo de geração de dados pelo ChatGPT. Esta tecnologia tem atraído uma atenção crescente e está a transformar fundamentalmente a criação de

## Composição de Equipas de Projetos de Desenvolvimento de Software com IA

---

conteúdos, capaz de criar dados sintéticos de alta qualidade, reduzir custos e tempo essencial para o treino de modelos de IA. Por isso esta tecnologia é considerada uma ferramenta indispensável para diversas aplicações tecnológicas com capacidades de gerar cenários de teste realistas, explorar novas hipóteses e facilitar na educação (Wang et al., 2023).

Neste contexto, para a modelação e teste dos algoritmos de *content matching* entre o projeto em avaliação e as competências dos colaboradores e entre o projeto em avaliação e projetos passados foram geradas através de interações com o ChatGPT dois conjuntos de dados (apêndice 1 e 2) com os seguintes atributos:

1. Conjunto com projetos: Nome; Descrição; Data de início; Data de fim; Equipas; ISP.
2. Conjunto com colaboradores: Nome; Experiências; e Competências;

### 4.3.2 Armazenamento dos dados gerados na base de dados

Após a criação dos dados no ChatGPT, os mesmos foram armazenados numa base de dados o SQLite para garantir a sua futura utilização pelo algoritmo criado em linguagem *Python* e pela aplicação *Web* a desenvolvida em *Flask* (Mönnich et al., 2024). A escolha do SQLite foi motivada por ser simples e fácil utilizar e de depurar no desenvolvimento de aplicações, para além de ter provado ser um motor de base de dados sólido e confiável ao longo da sua história (Allen & Owens, 2010).

Além do motor de gestão de base de dados SQLite foi também utilizada uma aplicação gráfica para manipulação das tabelas e dos dados designada DB Browser for SQLite (DB4S) (Mauricio Piacentini, 2024), uma interface gráfica de código aberto que permite interagir de forma eficiente e intuitiva com as bases de dados SQLite.

As figuras Figura 4.2 e Figura 4.1 e Figura 4.3 apresentam um extrato das tabelas “Projeto”, “Colaborador” e “Equipa” respetivamente, tal e qual como visualizado a partir do SQL Browser, que contém os dados sintéticos gerados com o ChatGPT.

## Composição de Equipas de Projetos de Desenvolvimento de Software com IA

Table: Projeto

idProjeto	nome	descricao_projeto	data_inicio	data_fim	equipa	isp
1	Desenvolvimento de aplicação Mobile	Criação de uma plataforma de comércio eletróni...	2024-11-01	2025-06-30	Equipa A: Desenvolvimento Software...	100 %...
2	Campanha de Marketing Digital	Implementação de uma campanha de marketing ...	2024-09-10	2025-01-15	Equipa B: Marketing Digital ...	90%
3	Software de Simulação de Processo Industrial	Criação de um software para simulação de ...	2024-09-10	2025-04-30	Equipa A: Desenvolvimento Software...	80%
4	Sistema de Gestão Escolar	Desenvolvimento de um sistema integrado para ...	2024-07-01	2024-04-26	Equipa A:Desenvolvimento de Software e ...	85%
5	Migração de Sistema ERP para Nuvem	Transferência de um sistema de ERP (Enterprise ...	2024-08-01	2025-03-15	Equipa A: Programadores	70%
6	Gestão orçamental	Planeamento e execução do lançamento de um ...	2024-08-15	2025-02-28	Equipa B: Marketing Digital...	60%
7	Aplicação de Gestão de Tarefas	Criação de uma aplicação para gestão de tarefas...	2024-07-15	2025-01-31	C: Gestores...	65%
8	Desenvolvimento de Aplicativo de Gestão de ...	Criação de um aplicativo para gerenciamento de ...	2024-07-15	2025-01-31	Equipa D: Consultoria Externa de UX/UI Design...	50%
9	Desenvolvimento de Plataforma de E-commerce	O projeto visa o desenvolvimento de uma ...	2024-11-01	2025-06-30	Equipa D: Consultoria Externa de UX/UI Design...	40%
10	Implementação de Sistema de Gestão Financeira	Desenvolvimento e implementação de um sistem...	2024-07-01	2025-01-31	Equipa A: Desenvolvimento Web...	20%
11	Sistema de Gestão de Orçamento Empresarial	Desenvolvimento de um sistema para gestão de ...	2024-09-01	2025-05-31	Equipa A:Desenvolvimento de Software e ...	10%
12	Reconstrução de software e design de sistemas	Reconstrução software empresarial, que inclui ...	2024-09-02	2024-09-30	Equipa H:Arquitetura de software e design de ...	95%

Figura 4.1. Extrato da tabela Projeto

Table: Colaborador

id	nome	competencias
1	João Silva	Desenvolvimento de aplicações móveis, Integraç...
2	Maria Santos	Desenvolvimento web, Base de dados, Algoritmo...
3	Pedro Oliveira	Desenvolvimento de aplicações móveis, Seguran...
4	Ana Costa	Desenvolvimento web, Algoritmos de otimização
5	Tiago	Desenvolvimento web, análise de dados, ...
6	Carlos Pereira	: Desenvolvimento de software, linguagens de ...
7	André Almeida	Finanças, análise de investimentos, contabilidade
8	Maria Costa	(Design gráfico, UX/UI design, branding)
9	Miguel Ferreira	Contabilista
10	Ricardo	Programador
11	Felipe Santos	Estratégias de marketing digital, SEO, PPC, análi...
12	Antonio ...	UX/UI design, prototipagem, usabilidade
13	Inês	Contabilidade financeira, gestão fiscal, análise d...
14	Adinilsa Pontes	Contabilidade financeira, planeamento tributário,...
15	Nilza	tratégias de marketing digital, SEO, gestão de ...

Figura 4.2. Extrato da tabela Colaborador

Table: Equipa

idEquipa	nome	designação
0	NULL	NULL
1	Equipa A	Desenvolvimento Software
2	Equipa B	Marketing Digital
3	Equipa C	Gestores
4	Equipa D	Consultoria Externa de UX/UI Design
5	Equipa E	Contabilista e Consultoria Fiscal
6	Equipa A	Programação em diversas linguagens (Java, ...
7	Equipa F	Conhecimento em base de dados e modelagem ...
8	Equipa G	Testes de software e garantia de qualidade
9	Equipa H	Arquitetura de software e design de sistemas
10	Equipa I	Gestão de tempo e capacidade de organização
11	Equipa K	Consultores para Resolver problemas e tomada ...
12	Equipa w	Gestores Informáticos
13	Equipa S	Contabilistas e Economistas
14	Equipa V	Programadores para aplicação Web
15	Equipa X	Programadores de aplicação Mobile

Figura 4.3. Extrato da tabela Equipa

Para além destas tabelas foi necessário criar tabelas de ligação para, por exemplo, saber que equipas, e consequentemente colaboradores, trabalharam em que projetos. Na Figura 4.4 apresenta-se o modelo Entidade-Relacionamento (ER) conceptual criado.

Composição de Equipas de Projetos de Desenvolvimento de Software com IA

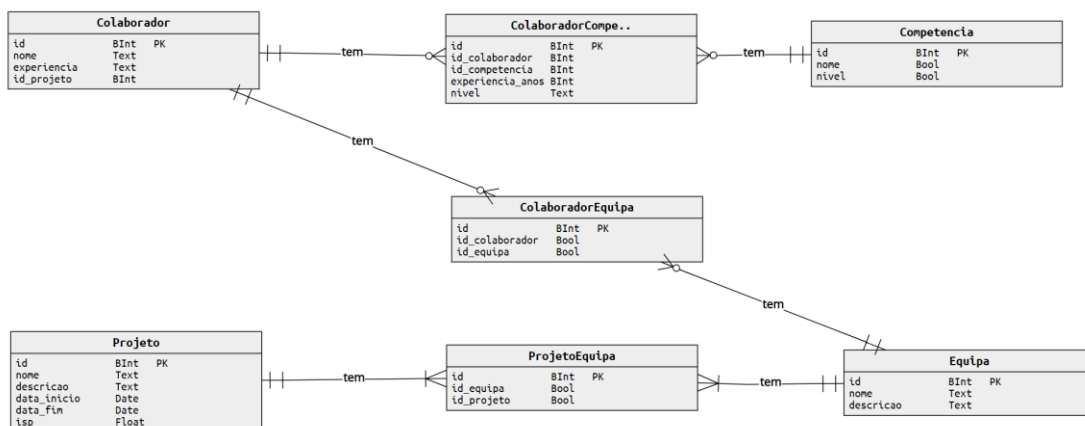


Figura 4.4. Modelo ER de suporte ao projeto criado na aplicação onda.dei.uc.pt

Após a geração e arquivo dos dados na base de dados SQLite os mesmos estão disponíveis para serem utilizados. No caso do trabalho apresentado neste relatório foi utilizada a linguagem *Python* para aceder aos dados tendo sido utilizada a biblioteca *sqlite3* do *Python* e funções associadas. Na Figura 4.6 mostra-se um extrato de código utilizado para aceder à base de dados SQLite3 *colaborador.db*. A função *conectar\_bd()* devolve uma ligação para a base de dados que pode ser posteriormente utilizada para a realização dos diferentes tipos de consultas (em inglês *queries*): *select*, *insert*, *update* e *delete*.

```
# Função para conectar ao banco de dados SQLite
def conectar_bd():
    conn = None
    try:
        conn = sqlite3.connect('colaborador.db')
        return conn
    except sqlite3.Error as e:
        print(e)
    return conn
```

Figura 4.5.Extrato de código com a função de conexão à base de dados

### 4.3.3 Pré-processamento de texto

O pré-processamento de dados é uma etapa crucial para garantir a qualidade e a utilidade dos dados, segundo a abordagem definida pelo Niranjan et al. (2016) o pré-processamento permite tratar os dados do texto isto inclui a tokenização (permite converter tudo em minúsculas, a remoção de *stop words*, lematização (em inglês, *lemmatization* ou

Composição de Equipas de Projetos de Desenvolvimento de Software com IA

*stemming*), remoção de pontuações e caracteres especiais, correção de erros e vetorização.

Para a implementação do pré-processamento, foram importadas as bibliotecas do *Natural Language Toolkit* (NLTK), na ferramenta de desenvolvimento o *Python*, que oferece uma variedade de ferramentas e métodos para processar e analisar texto em linguagem natural, visualizar na Figura 4.6.

```
import nltk
from nltk.tokenize import word_tokenize
from nltk.corpus import stopwords
from nltk.stem import WordNetLemmatizer
```

Figura 4.6. Extrato de código Python com as bibliotecas importadas para o pré-processamento de texto

De seguida, procedeu-se à primeira etapa do pré-processamento de dados, a limpeza de dados que envolve a remoção de qualquer informação irrelevante ou redundante dos dados, como caracteres especiais, pontuação, dígitos e *stopwords*, bem como converter o texto para minúsculas e remover dados duplicados, visualizar na Figura 4.7.

Texto original: Desenvolvimento de aplicações móveis, Integração com APIs externas, Segurança de dados.  
Tokens: ['desenvolvimento', 'aplicações', 'móveis', 'integração', 'apis', 'externas', 'segurança', 'dado']

Figura 4.7. Texto após a tokenização

Como se pode ver Figura 4.7 o texto é dividido em unidades menores, chamadas *tokens* (“Desenvolvimento”; “Aplicações”, “Móveis”). Os *tokens* são geralmente palavras individuais ou partes de palavras, separadas com base em regras específicas (como espaços em branco ou pontuações) que podem ser facilmente manipuladas e analisadas.

Texto original: Desenvolvimento de aplicações móveis, Integração com APIs externas, Segurança de dados.  
Texto pré-processado: desenvolvimento aplicações móveis integração apis externas segurança dado

Figura 4.8. Texto após o pré-processamento

Após a tokenização (ver Figura 4.8), o texto passa por várias etapas de pré-processamento, que podem incluir remoção de *stopwords* (palavras irrelevantes para reduzir às suas formas básicas) permite tratar os dados, limpar e estruturá-los para a normalização do texto. Foram removidas algumas palavras da análise consideradas como *stop* (“de”; “Com” e os “dados”).

Composição de Equipas de Projetos de Desenvolvimento de Software com IA

A terceira etapa do pré-processamento de dados é a tokenização e a lematização. A Figura 4.9 apresenta a sua implementação. O pré-processamento do texto ajuda garantir as características extraídas do texto para torná-las consistentes, relevantes para uma análise eficaz dos modelos de NLP.

```
def preprocess_text(text):
    tokens = word_tokenize(text.lower()) # Tokenização e conversão para minúsculas
    tokens = [lemmatizer.lemmatize(token) for token in tokens if token not in stop_words and token
              | | | not in punctuation] # Lemmatização e remoção de stop words e pontuações
    return tokens

texto_exemplo = "Desenvolvimento de aplicações móveis, Integração com APIs externas, Segurança de dados."
tokens = preprocess_text(texto_exemplo)
print("Texto original:", texto_exemplo)
print("Tokens:", tokens)
```

Figura 4.9. Método de pré-processamento

Por fim, a vectorização transforma texto em uma representação numérica, através da representação de texto com TF-IDF. Ao realizar a lematização e a remoção de *stop words* na língua portuguesa permite adaptar o pré-processamento para as características específicas do idioma, o que pode resultar em melhores resultados em tarefas que envolvem textos em português.

A técnica TF-IDF é utilizada para representar a importância relativa de uma palavra em um documento dentro de uma coleção de documentos. É calculado multiplicando duas métricas:

- Frequência do Termo (TF): Mede a frequência com que uma palavra ocorre em um documento. É calculada como a contagem de ocorrências de uma palavra dividida pelo número total de palavras no documento. Isso ajuda a capturar a importância da palavra dentro do documento.
- Frequência Inversa do Documento (IDF): Mede a raridade de uma palavra em toda a coleção de documentos. É calculada como o logaritmo do inverso da fração de documentos que contêm a palavra. Palavras frequentes em muitos documentos terão um IDF mais baixo, enquanto palavras raras terão um IDF mais alto.

O texto passa por um processo de pré-processamento, em seguida é calculada o TF-IDF a partir das competências dos funcionários com o método *fit\_transform*, que converte uma coleção de documentos de texto em uma matriz TF-IDF. A descrição do projeto é

## Composição de Equipas de Projetos de Desenvolvimento de Software com IA

---

transformada num vetor TF-IDF pelo mesmo *vetorizador*. Isso é feito na função *comparar\_projeto*.

```
# Função para realizar pré-processamento no texto
def preprocessamento(texto):
    lemmatizer = WordNetLemmatizer()
    stop_words = set(stopwords.words('portuguese'))
    tokens = word_tokenize(texto.lower())
    tokens = [lemmatizer.lemmatize(token) for token in tokens if token not in stop_words]
    return ' '.join(tokens)
```

Figura 4.10. Pré-processamento de Texto para língua portuguesa

O processo do pré-processamento permite preparar os dados e transformá-los em vetores TF-IDF usando o método *fit\_transform para comparar\_projeto*. Para isso, o *vetorizador* (fit) é ajustado com as competências dos colaboradores e com documentos de descrição de projeto para os transformar em vetores de TF-IDF.

## 4.4 Modelação

Nesta seção, apresentam-se dois algoritmos de *content matching* desenvolvidos utilizando técnicas de NLP e de ItemCF: o primeiro algoritmo procura identificar com base na descrição do projeto quais os melhores profissionais para a execução do mesmo; e o segundo algoritmo procura por projetos similares através da descrição dos mesmos, de forma a que o gestor de projetos possa avaliar qual a equipa de profissionais mais adequada ao projeto, que terá por base a similaridade e o ISP. Ambos os algoritmos utilizaram a similaridade do cosseno, como proposto por Rodrigues (2023). Dado que ambos os algoritmos utilizaram a similaridade do cosseno apresenta-se a mesma numa primeira secção, sendo depois apresentados os dois algoritmos e, por fim, a aplicação gráfica de interface de utilizador criada para a utilização dos mesmos.

### 4.4.1 Métrica similaridade do cosseno

A similaridade de cosseno é uma métrica utilizada para medir a similaridade entre dois vetores, sendo particularmente útil para comparar documentos ou outros pontos de dados num espaço multidimensional (Hussain et al., 2021; Murugesan et al., 2010).

Composição de Equipas de Projetos de Desenvolvimento de Software com IA

Utilizando a função *cosine\_similarity* da biblioteca *scikit-learn* do *Python* é possível calcular esta métrica. Na Figura 4.11 apresenta-se um exemplo do cálculo da mesma em que é possível constatar que vetores com valores diferentes nas mesmas posições o valor é 0 e para vetores com elementos iguais nas mesmas posições o valor é 1 (ex. a posição [0,1] da matriz de similaridade que representa o cálculo da similaridade entre X[0] e Y[1] que são iguais tem o valor 1).

```
#Exemplo do cálculo da similaridade de cosseno
import numpy as np
from sklearn.metrics.pairwise import cosine_similarity
# Vetores X and Y
matriz_1 = np.array([[1, 0, -1], [0, 1, 0]])
matriz_2 = np.array([[1, 1, 0], [1, 0, -1]])
# Cálculo da similaridade do cosseno
matriz_de_similaridade = cosine_similarity(matriz_1, matriz_2)
print(matriz_de_similaridade)
```

[23] ✓ 0.0s Open 'matriz\_de\_similaridade' in Data Wrangler

```
... [[0.5      1.      ]
     [0.70710678 0.      ]]
```

Figura 4.11. Exemplo do cálculo da métrica similaridade do cosseno em Python

A semelhança de cosseno varia entre -1 (exatamente o oposto) e 1 (exatamente o mesmo), com 0 a indicar ortogonalidade (nenhuma semelhança).

#### 4.4.2 Algoritmo de *content matching* das competências

O algoritmo de *content matching* das competências compara a descrição do projeto com as descrições características dos colaboradores. Para tal, foi criado um *script* em *Python*, que primeiro acede à base de dados *colaborador.db* criada no *SQLite* para devolver o conjunto com os colaboradores e respetivas descrições das competências. Posteriormente, utiliza as técnicas de pré-processamento de texto identificadas na secção 4.3.3. para preparar os textos a comparar. E, por fim, calcula a similaridade devolvendo os colaboradores cujas competências tenham uma similaridade superior a 0,5 à descrição do projeto. A Figura 4.12 apresenta o extrato de código relativo à função *comparar\_projeto(descrição\_projeto, lista)* responsável pela execução das tarefas descritas.

Composição de Equipas de Projetos de Desenvolvimento de Software com IA

```
# Função para comparar a descrição do projeto atual com os projetos anteriores e retornar os funcionários que trabalharam ne
def comparar_projeto(descricao_projeto, lista):
    vectorizer = TfidfVectorizer()
    descricao_projeto_tfidf = vectorizer.fit_transform([descricao_projeto])
    resultados = []
    projeto_encontrado = False # Variável para controlar se o projeto foi encontrado na base de dados

    for linha_colaborador in lista:
        # Pré-processamento da descrição do colaborador anterior
        projeto_preprocessado = preprocessamento(linha_colaborador[1])
        # Transforma a descrição das competências anterior em um vetor TF-IDF
        competencias_tfidf = vectorizer.transform([projeto_preprocessado])
        # Calcula a similaridade entre a descrição do projeto e as competências
        similaridades = cosine_similarity(descricao_projeto_tfidf, competencias_tfidf)

        # Se a similaridade for alta o suficiente, adiciona os colaboradores aos resultados
        if similaridades[0][0] > 0.5:
            resultados.append( [linha_colaborador[0], linha_colaborador[1], linha_colaborador[2], linha_colaborador[3],
                                linha_colaborador[4], similaridades[0][0]] )
            projeto_encontrado = True
            resultados = sorted(resultados, key=lambda x: x[5], reverse=True)

    if not projeto_encontrado: # Se o projeto não foi encontrado, emitir uma mensagem de erro
        flash('Competências não encontradas na base de dados.')

    return resultados
```

Figura 4.12. Extrato do código responsável pelo cálculo da similaridade entre a descrição do projeto e das descrições das competências dos colaboradores

MATCHING DAS COMPETÊNCIAS

Desenvolvimento de aplicações móveis

Comparar
Limpar

**Resultados:**

**Colaborador 1 : João Silva**

- **Competências:** Desenvolvimento de aplicações móveis, Integração com APIs externas, Segurança de dados
- Projetos Desenvolvidos:** 5
- Experiência:** João Silva é um desenvolvedor de software experiente, com mais de 8 anos de experiência no desenvolvimento de aplicativos móveis para diversas plataformas. Ele possui um amplo conhecimento em integração com APIs externas e é especialista em garantir a segurança dos dados dos usuários.
- Equipas:** Equipe A: Desenvolvimento Software Equipe B: Marketing Digital Equipe D: Consultoria Externa de UX/UI Design
- Similaridade:** 1.0000000000000002

**Colaborador 2 : Pedro Oliveira**

- **Competências:** Desenvolvimento de aplicações móveis, Segurança de dados
- Projetos Desenvolvidos:** 3
- Experiência:** Pedro Oliveira é um desenvolvedor de aplicativos móveis talentoso, com experiência em criar soluções inovadoras para dispositivos móveis. Ele tem um foco especial em garantir a segurança dos dados do usuário e implementar medidas robustas de proteção de dados.
- Equipas:** Equipe D: Consultoria Externa de UX/UI Design Equipe B: Marketing digital
- Similaridade:** 1.0000000000000002

**Colaborador 3 : Maria Santos**

- **Competências:** Desenvolvimento web, Base de dados, Algoritmos de otimização
- Projetos Desenvolvidos:** 7
- Experiência:** Maria Santos é uma desenvolvedora web altamente qualificada, com habilidades sólidas em criação de interfaces web interativas e eficientes. Ela é especializada em design de bancos de dados e tem experiência com algoritmos de otimização para melhorar o desempenho de sistemas online.
- Equipas:** Equipe A: Desenvolvimento de Software Equipe D: Consultoria Externa de UX/UI Design
- Similaridade:** 0.5773502691896258

**Colaborador 4 : Ana Costa**

- **Competências:** Desenvolvimento web, Algoritmos de otimização
- Projetos Desenvolvidos:** 5
- Experiência:** Ana Costa é uma desenvolvedora web apaixonada por criar experiências online excepcionais. Ela tem uma ampla experiência em desenvolvimento web e é especialista em otimizar algoritmos para melhorar o desempenho e a eficiência de aplicativos e sites.
- Equipas:** Equipe V: Programadores para aplicação Web Equipe X: Programadores de aplicação Mobile
- Similaridade:** 0.5773502691896258

Figura 4.13. Resultado da comparação da descrição de um projeto com as competências dos colaboradores existentes na base de dados

## Composição de Equipas de Projetos de Desenvolvimento de Software com IA

Na Figura 4.13 é possível ver o resultado da comparação da descrição de um projeto com as competências dos colaboradores existentes na base de dados. Para este projeto em concreto, cuja descrição é “Desenvolvimento de aplicações móveis”, foram devolvidos quatro colaboradores, os dois primeiros com uma similaridade de 1 e os dois últimos com uma similaridade de 0.577. Os colaboradores foram ordenados decendentemente pelo valor da similaridade.

### 4.4.3 Algoritmo de *content matching* dos projetos

O algoritmo de *content matching* dos projetos envolve comparação entre as descrições do projeto que se pretende realizar e todas as descrições passadas dos projetos realizados que estão armazenadas na base de dados, passando pelos mesmos passos de pré-processamento, criação da matriz TD-IDF e cálculo da similaridade. A Figura 4.14 apresenta o extrato do código que faz estes passos no caso específico dos projetos.

```
# Função para comparar a descrição do projeto atual com os projetos anteriores e retornar os funcionários que trabalharam nesse
def comparar_projeto(descricao_projeto, lista):
    vectorizer = TfidfVectorizer()
    descricao_projeto_tfidf = vectorizer.fit_transform([descricao_projeto])

    resultados = []
    projeto_encontrado = False # Variável para controlar se o projeto foi encontrado na base de dados

    print(lista)
    for linha_projeto in lista:
        # Pré-processamento da descrição do projeto anterior
        projeto_preprocessado = preprocessamento(linha_projeto[1])
        # Transforma a descrição do projeto anterior em um vetor TF-IDF
        projeto_tfidf = vectorizer.transform([projeto_preprocessado])
        # Calcula a similaridade entre a descrição do projeto atual e a descrição do projeto anterior
        similaridades = cosine_similarity(descricao_projeto_tfidf, projeto_tfidf)

        # Se a similaridade for alta o suficiente, adiciona os funcionários associados ao projeto aos resultados
        if similaridades[0][0] > 0.5:
            resultados.append( [linha_projeto[0], linha_projeto[1], linha_projeto[2], linha_projeto[3], similaridades[0][0]])
            projeto_encontrado = True

    if not projeto_encontrado: # Se o projeto não foi encontrado, emitir uma mensagem de erro
        flash('Projeto não encontrado na base de dados.')

    return resultados
```

Figura 4.14. Cálculo da similaridade entre descrições de projetos

Após este cálculo são retornados, numa lista, todos os projetos cuja similaridade é superior a 0.5. Para além do nome do projeto são também devolvidas outras informações como a descrição, equipas participantes, taxa de similaridade e taxa de sucesso dos projetos para apoiar o gestor de projetos na tomada de decisão, pois mesmo que haja projetos com similitudes maiores o mesmo pode optar por escolher um projeto passado

## Composição de Equipas de Projetos de Desenvolvimento de Software com IA

de maior sucesso. Na Figura 4.15 apresenta-se o *output* na aplicação *web* dos resultados da pesquisa, onde é possível constatar que o projeto mais similar (0.845...) é também o projeto com maior sucesso (90%), pelo que poderão ser as equipas que trabalharam neste aquelas que irão trabalhar no novo projeto.

MATCHING DOS PROJETOS

Plataforma web de uma campanha de marketing digital para lançamento de uma aplicação.

Comparar
Limpar

**Resultados:**

**Projeto 1: Campanha de Marketing Digital**

- Equipa: Equipa B: Marketing Digital Equipa A: Desenvolvimento de Software Equipa D: Consultoria Externa de UX/UI Design
- Índice de Sucesso do Projeto: 90%
- Similaridade: 0.8451542547285168

**Projeto 2: Desenvolvimento de Plataforma de E-commerce**

- Equipa: Equipa D: Consultoria Externa de UX/UI Design Equipa B: Marketing digital
- Índice de Sucesso do Projeto: 40%
- Similaridade: 0.5070925528371101

Figura 4.15. Resultado da comparação da descrição de um projeto com os projetos existentes na base de dados

### 4.4.4 Aplicação gráfica de interface com o utilizador

Nesta secção, apresenta-se a aplicação de interface gráfica do utilizador (GUI, do inglês *graphical user interface*) desenvolvida para interagir com os algoritmos desenvolvidos. Esta aplicação foi desenvolvida com recurso à linguagem de programação *Python*, às linguagens associadas ao desenvolvimento de aplicações *web* como a *Hypertext Markup Language* (HTML) e à *framework Flask* (Mönnich et al., 2024), uma *micro framework desenvolvida em Python* de fácil utilização para criação de pequenas aplicações *web*. Adoção da linguagem *Python* deve-se essencialmente a dois motivos: facilidade da estudante com a mesma; e facilidade de integração com os algoritmos de IA, comumente em *Python*, como foi o caso dos desenvolvidos neste trabalho.

Além disso, o desenvolvimento do GUI envolveu a utilização da base de dados SQLite, já referida anteriormente. Este recurso revelou-se crucial para a gestão eficiente dos dados, proporcionando uma estrutura organizada e acessível para o armazenamento e manipulação das informações necessárias ao funcionamento do algoritmo.

## Composição de Equipas de Projetos de Desenvolvimento de Software com IA

A aplicação *web* desenvolvida, para além de suportar a execução dos algoritmos criados em *Python*, permite também a atualização das tabelas existentes na base de dados.

A *framework Flask* funciona através da definição de rotas (*endpoints*) para a execução de tarefas específicas. A título de exemplo, apresenta-se de seguida a forma como é feito o processamento do formulário para inclusão da descrição do projeto no caso do algoritmo de comparação de projetos, que corresponde à rota “/”, que representa tipicamente a rota de entrada numa aplicação *web*.

A informação é introduzida pelo utilizador através do formulário com a rota *Flask* para lidar com o pedido submetido (Figura 4.16).

```
@app.route('/', methods=['GET', 'POST'])
def index():
    resultados = None
    if request.method == 'POST':
        descricao_projeto = preprocessamento(request.form['descricao_projeto'])
        print("Descrição do projeto após pré-processamento:", descricao_projeto) # Depuração
        resultados = comparar_projeto(descricao_projeto)
        print("Resultados:", resultados) # Depuração
    return render_template('index.html', resultados=resultados)

if __name__ == '__main__':
    app.run(debug=True)
```

Figura 4.16 Rota para processamento do formulário do algoritmo da comparação da descrição do projetos

Nesta rota, quando um pedido é enviado para a página inicial (“/”), o *Flask* verifica se é uma solicitação POST (ou seja, se o formulário foi submetido). Se for uma solicitação POST, o *Flask* obtém a descrição do projeto do formulário, pré-processa o texto e chama a função *comparar\_projeto()* para obter os resultados da comparação. Por fim, apresenta uma página *web* que é o resultado da *layout* definido no *template index.html* (Figura 4.17) e dos resultados obtidos da função *comparar\_projeto()*, cujo resultado final pode ser visto na Figura 4.15. De referir ainda que o *template index.html* contém o formulário para submeter o texto da descrição do projeto a comparar, bem como, o tratamento da apresentação dos resultados.

Composição de Equipas de Projetos de Desenvolvimento de Software com IA

```

<div class="container">
  <div class="form-container">
    <form action="/" method="post" onsubmit="return validateForm()">
      <textarea class="form-input" id="descricao_projeto" name="descricao_projeto" rows="4" cols="50"
    <div id="notification" class="notification">
      {% with messages = get_flashed_messages() %}
      {% if messages %}
        {{ messages[0] }}
      {% endif %}
      {% endwith %}
    </div>
    <input class="form-submit" type="submit" value="Comparar" id="submit-btn">
    <button type="button" onclick="clearResult()">Limpar</button>
  </form>
</div>
<div class="result-container">
  <div id="resultados">
    {% if resultados %}
      <div class="result-text">
        <h2>Resultados:</h2>
        {% set i = 1%}
        {% for projeto in resultados%}
          <h3>Projeto {{ loop.index }}: {{ projeto[0] }}</h3>
          <ul>
            <li>
              <strong>Equipa:</strong> {{ projeto[2] }}<br>
            </li>
            <li>
              <strong>Índice de Sucesso do Projeto:</strong> {{ projeto[3] }}<br>
            </li>
            <li>
              <strong>Similaridade:</strong> {{ projeto[4] }}<br>
            </li>
          </ul>
        {% endfor %}
      </div>
    {% endif %}
  </div>
</div>

```

Figura 4.17. Extrato do código do template para a apresentação do algoritmo de comparação da descrição dos projetos

O formulário permite ao utilizador preencher com informações da descrição do projeto, e com o botão “Compare”, enviar a informação através do método POST para a rota da própria página. Por fim, é chamada a função *validateForm()* (Figura 4.18), para garantir que o campo de descrição do projeto não esteja vazio antes de enviar a informação.

Composição de Equipas de Projetos de Desenvolvimento de Software com IA

```

</div>
</div>

<script>
function validateForm() {
    var descricaoProjeto = document.getElementById("descricao_projeto").value;
    var notification = document.getElementById("notification");
    if (descricaoProjeto.trim() === "") {
        notification.innerText = "Por favor, escreva algum texto antes de comparar.";
        return false;
    }
    notification.innerText = "";
    return true;
}

```

Figura 4.18. Extrato do código do template para a apresentação do algoritmo de comparação da descrição dos projetos que contém a função de javascript de validação do formulário

Na Figura 4.19 é possível visualizar a mensagem de alerta sempre que o utilizador tentar submeter o formulário com o campo da descrição do projeto vazio. A impossibilidade submeter o formulário vazio obriga o utilizador a introduzir os dados antes de submeter o formulário.

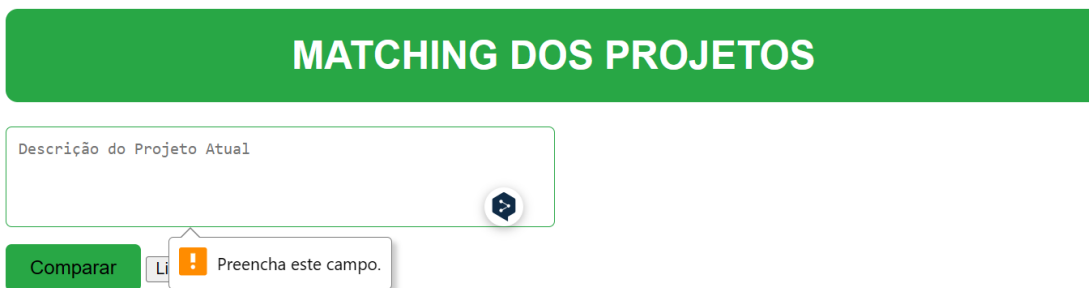


Figura 4.19. Mensagem de aviso de descrição do projeto vazio

A mensagem “Projeto não encontrado na base de dados” é exibida para informar que a busca não retornou nenhum resultado correspondente. Isso pode acontecer por várias razões, como o projeto sendo único ou muito diferente dos projetos previamente registados na base de dados. Isto ocorre após percorrer todos os projetos na base de dados e calcular a similaridade entre as descrições, o código verifica se a variável *projeto\_encontrado* for falso, indica que nenhum projeto semelhante foi encontrado e emite a mensagem.

```

# Se a similaridade for alta o suficiente, adiciona os funcionários associados ao projeto
if similaridades[0][0] > 0.5:
    resultados.append( [linha_projeto[0], linha_projeto[1], linha_projeto[2], linha_projeto[3], linha_projeto[4]] )
    projeto_encontrado = True

if not projeto_encontrado: # Se o projeto não foi encontrado, emitir uma mensagem de erro
    flash('Projeto não encontrado na base de dados.')

return resultados

```

Figura 4.20. Função que emite a mensagem quando o projeto não é encontrado

A mensagem de erro é emitida, através da função *flash()*, em seguida informa ao utilizador que o projeto não foi encontrado na base de dados e é exibida na interface do utilizador.

## 4.5 Avaliação

Nesta etapa o objetivo era o de avaliar os algoritmos numa empresa de desenvolvimento de software, através da introdução dos textos da descrição e/ou caderno de encargos do projeto, o que não foi possível. Não obstante, procurou-se na internet descrição de projetos de desenvolvimento de *software* para testar os algoritmos com casos reais de projetos de desenvolvimento de software (anexos I e II), ambos da empresa GSTEP.

A GSTEP© (2020) é uma empresa de excelência em *Business Intelligence (BI)*, *Enterprise Performance Management (EPM)* e *Business Analytics (BA)*, que opera nos mercados de Portugal, Europa e África e nos setores dos Seguros, Saúde, Logística, Banca, Retalho, Telecomunicações, Administração Pública e Indústria, possuindo uma equipa sénior e qualificada. É líder em qualidade e entrega de projetos, sustentada pelos seus valores de Confiança, Competência e Compromisso. Presentes em Portugal (sede), Espanha e Moçambique, estão estrategicamente posicionados para servir os setores dos Seguros, Saúde, Logística, Banca, Retalho, Telecomunicações, Administração Pública e Indústria.

Nas subsecções seguintes apresentam-se os testes realizados aos dois algoritmos com os textos dos casos de sucesso da GSTEP que se encontram nos anexos I e II.

#### 4.5.1 Teste do algoritmo de *content matching* dos projetos

Para este primeiro teste utilizou-se o caso de sucesso descrito no anexo I, o do construção de uma ferramenta de *software* de suporte ao Planeamento e Controlo de Gestão na empresa.

A descrição do caso de sucesso conforme o anexo I é a seguinte:

*“A Companhia de Seguros internacional, implantada em Portugal e líder no seu setor de mercado, fazia todo o planeamento e controlo orçamental através de folhas de cálculo Excel que, numa empresa desta dimensão, levantava os seguintes problemas:*

- *Esforço de cerca de 80% no tratamento de dados, tendo apenas 20% para análise dos mesmos*
- *Dificuldade na criação de diversos cenários de orçamento*
- *Gestão de versões de ficheiros de Excel*
- *Informação centralizada no departamento de Planeamento e Controlo de Gestão, sendo que sempre que alguma área de negócio precisava de analisar os desvios tinha de esperar para receber essa informação tratada*
- *Risco de perder informação caso algum ficheiro fosse danificado*
- *Risco de falhas de segurança*
- *Por forma a reduzir o esforço no tratamento de dados e aumentar o tempo para analisar resultados, indicadores e tendências, a Seguradora optou por automatizar o processo de forma a colocar todo o know-how num software user-friendly.”*

A solução proposta pela GSTEP para o projeto do sistema de Planeamento e Controlo de Gestão que permitisse a introdução do orçamento da Seguradora, a extração de dados para *reporting* externo e a produção de relatórios para controlo orçamental, foi a implementação de uma ferramenta de *Enterprise Performance Management (EPM)*, a Oracle Hyperion. Este projeto inclui várias etapas como o levantamento de requisitos, conceção e implementação do EPM, com o desenvolvimento de formulários para inserção de dados e sistemas analíticos para *reporting*, tendo havido ainda lugar para a formação dos utilizadores no novo sistema.

## Composição de Equipas de Projetos de Desenvolvimento de Software com IA

Para a utilização do protótipo desenvolvido colocou-se a seguinte descrição para o projeto: “Desenvolvimento de um sistema de *enterprise performance management* com o Oracle Hyperion de suporte ao controlo de gestão sendo necessária a realização da análise de requisitos, desenvolvimento de formulários, sistema analítico de *reporting* e formação aos utilizadores.”

MATCHING DOS PROJETOS

Desenvolvimento de um sistema de *enterprise performance management* com o Oracle Hyperion de suporte ao controlo de gestão sendo necessária a realização da análise de requisitos, desenvolvimento de formulários, sistema analítico de *reporting* e formação aos utilizadores

●

Comparar
Limpar

**Resultados:**

**Projeto 1: Sistema de Informação de Gestão para o Controlo de Gestão**

- **Equipa:** Equipa X: Analistas funcionais; Equipa Y: Equipa Low Code; Equipa Z: Equipa de Formação.
- **Índice de Sucesso do Projeto:** 90%
- **Similaridade:** 0.6240377207533829

**Projeto 2: Aplicação Comercial**

- **Equipa:** Equipa V: Programadores para aplicação Web, Equipa X: Programadores de aplicação Mobile.
- **Índice de Sucesso do Projeto:** 88%
- **Similaridade:** 0.5883484054145522

**Projeto 3: Sistema de Gestão Escolar**

- **Equipa:** Equipa A: Desenvolvimento de Software e Tecnologia, Equipa W: Gestores Informáticos, Equipa H: Arquitetura de software e design de sistemas, Equipa E: Contabilista e Consultoria Fiscal
- **Índice de Sucesso do Projeto:** 85%
- **Similaridade:** 0.5661385170722981

**Projeto 4: Sistema de Informação de Gestão de Cantinas**

- **Equipa:** Equipa X: Analistas funcionais; Equipa Y: Equipa Low Code;
- **Índice de Sucesso do Projeto:** 80%
- **Similaridade:** 0.6201736729460425

**Projeto 5: Implementação de Sistema de Gestão Financeira**

- **Equipa:** Equipa A: Desenvolvimento Web, Equipa V: Programadores para aplicação Web, Equipa X: Programadores de aplicação Mobile,
- **Índice de Sucesso do Projeto:** 20%
- **Similaridade:** 0.5661385170722981

Figura 4.21. Teste do algoritmo de content matching dos projetos

É possível constatar pela Figura 4.21 que para o texto introduzido foram devolvidos vários projetos de diferentes naturezas, equipas participantes, níveis de sucesso e índices de similaridade, podendo o gestor de projeto analisar a lista de projetos desenvolvidos e avaliar quais as equipas a escolher para o novo projeto.

### 4.5.2 Teste do algoritmo de *content matching* das competências

À semelhança do que foi feito para o algoritmo anterior transcreve-se aqui a descrição do segundo caso também disponível no anexo II:

*“A companhia de Seguros utilizava uma tecnologia em descontinuação, decidindo avançar com uma prova de conceito de forma a selecionar a tecnologia e o parceiro implementador que melhor respondiam às suas necessidades atuais*

*Composição de Equipas de Projetos de Desenvolvimento de Software com IA*

---

*e futuras, entre as quais, aumentar a performance e a usabilidade da ferramenta de suporte à decisão.”*

A solução proposta pela GSTEP para responder aos objetivos apresentados pela Lusitania foi a criação de uma plataforma para oferecer uma ampla gama de cursos em diversas áreas, incluindo tecnologia da informação, negócios e design. O objetivo é criar uma experiência de aprendizagem personalizada e interativa, com recursos como vídeos a pedido, sessões ao vivo, fóruns de discussão e avaliações automatizadas. A plataforma também deve ser acessível em dispositivos móveis, proporcionando aos utilizadores a flexibilidade de aprender a qualquer hora e em qualquer lugar. O desafio deste projeto inclui a integração de tecnologias de última geração, como aprendizado de máquina para recomendação de cursos personalizados, análise de dados para avaliar o progresso do aluno e sistemas de segurança robustos para proteger a privacidade dos dados dos utilizadores.

Para a utilização do protótipo desenvolvido colocou-se a seguinte descrição para o projeto: “Desenvolvimento de uma plataforma de web de e-learning com sistema de recomendação de cursos.”.

A Figura 4.22 apresenta os resultados da pesquisa que retornou quatro colaboradores possíveis para trabalhar neste projeto, ordenados descendentemente por similaridade das competências com a descrição do projeto. Este algoritmo devolve os colaboradores, competências, projetos desenvolvidos em que participou, experiência passada e as equipas de que faz ou fez parte.

Estes resultados auxiliam o gestor de projeto na tomada de decisão que de outra forma teria de escolher os colaboradores de memória o que poderia levar a erros como a se esquecer de algum colaborador importante para o projeto dadas as suas competências.

## Composição de Equipas de Projetos de Desenvolvimento de Software com IA

MATCHING DAS COMPETÊNCIAS

Desenvolvimento de uma plataforma de web de e-learning com sistema de recomendação de cursos

Comparar
Limpar

**Resultados:**

**Colaborador 1 : Antonio Trigo**

- **Competências:** Desenvolvimento de sistemas de recomendação e de plataformas web de e-learning.
- Projetos Desenvolvidos:** 5
- Experiência:** 20 anos no desenvolvimento de software, pelo menos :)
- Equipas:** Equipa V:Programadores para aplicação Web Equipa X:Programadores de aplicação Mobile Equipa F: Programadores de IA
- Similaridade:** 0.7559289460184546

**Colaborador 2 : Maria Santos**

- **Competências:** Desenvolvimento web, Base de dados, Algoritmos de otimização
- Projetos Desenvolvidos:** 7
- Experiência:** Maria Santos é uma desenvolvedora web altamente qualificada, com habilidades sólidas em criação de interfaces web interativas e eficientes. Ela é especializada em design de bancos de dados e tem experiência com algoritmos de otimização para melhorar o desempenho de sistemas online.
- Equipas:** Equipe A: Desenvolvimento de Software Equipa D: Consultoria Externa de UX/UI Design
- Similaridade:** 0.5345224838248489

**Colaborador 3 : Ana Costa**

- **Competências:** Desenvolvimento web, Algoritmos de otimização
- Projetos Desenvolvidos:** 6
- Experiência:** Ana Costa é uma desenvolvedora web apaixonada por criar experiências online excepcionais. Ela tem uma ampla experiência em desenvolvimento web e é especialista em otimizar algoritmos para melhorar o desempenho e a eficiência de aplicativos e sites.
- Equipas:** Equipa V:Programadores para aplicação Web Equipa X:Programadores de aplicação Mobile
- Similaridade:** 0.5345224838248489

**Colaborador 4 : Tiago**

- **Competências:** Desenvolvimento web, análise de dados, gerenciamento de projet
- Projetos Desenvolvidos:** 4
- Experiência:** Experiência: 5 anos em desenvolvimento web e análise de dados.
- Equipas:** Equipe A: Desenvolvimento Web Equipa V:Programadores para aplicação Web Equipa X:Programadores de aplicação Mobile
- Similaridade:** 0.5345224838248489

Figura 4.22. Teste do algoritmo de content matching das competências

## 4.6 Entrega

Este ponto da metodologia não a chegou a ser realizado por não existir uma empresa em que fosse possível testar os algoritmos desenvolvidos, algo que se espera vir a fazer no âmbito dos trabalhos futuros associados.

## CONCLUSÃO

Este capítulo visa sintetizar o trabalho realizado, rever as questões de pesquisa que fundamentaram a revisão da literatura, refletir sobre os resultados obtidos, bem como, identificar as limitações encontradas ao longo do trabalho realizado e propor direções para trabalhos futuros.

O tema proposto visa apresentar a utilização da IA na composição de equipas em projetos de desenvolvimento de *software*. O estudo começou por contextualizar a evolução e a aplicação da IA em diversos domínios, destacando as várias técnicas que contribuem no ramo da IA para a GP, conforme referido por Taboada et al. (2023). Técnicas essas que interligam conceitos desde a otimização de recursos até a alocação de tarefas e tomada de decisão, utilizando algoritmos de *Machine Learning (ML)*, *Fuzzy Logic (FL)*, *Neural Networks (NN)* e *Natural Language Processing (NLP)*. Os resultados do Taboada et al., (2023), demonstraram a importância da IA como uma ferramenta poderosa que auxilia em diversos setores da sociedade, e que incorpora algoritmos avançados para proporcionar benefícios para os gestores de projetos. Ao longo do processo da revisão de literatura, foi possível identificar que a Inteligência Artificial está presente em diversas áreas como saúde, segurança, serviços bancários e financeiros, pois a sua contribuição é valiosa para otimizar o processo de desenvolvimento de software nas organizações.

Após uma revisão dos conceitos teóricos de suporte ao desenvolvimento de sistemas de IA similares ao do trabalho a desenvolver realizou-se uma RSL com o intuito de descobrir o que estava a ser feito neste domínio, tendo-se identificado 10 estudos (Tabela 2.1).

De seguida realizou-se a experiência de aplicação das tecnologias de IA com o desenvolvimento de um trabalho de ciência de dados de acordo com a metodologia CRISP-DM: entendimento do negócio, em que se apresentou brevemente o que é a composição de equipas de projetos de desenvolvimento de software; entendimento dos dados, em que se apresentou a caracterização dos dados são necessários para o trabalho; preparação dos dados, desde à geração dos mesmos à vectorização; modelação, onde se apresentaram os algoritmos e a app desenvolvida; avaliação, com o teste não real dos

## Composição de Equipas de Projetos de Desenvolvimento de Software com IA

---

algoritmos desenvolvidos; e entrega, que não foi realizada por não existir um ambiente real.

Em termos de contributos da presente dissertação, para além da sistematização dos conceitos associados ao desenvolvimento de sistemas similares, foram identificadas outras utilizações da IA neste domínio e, como principal contributo, a proposta dos dois algoritmos de *content matching* que utilizam técnicas de NLP e ItemCF.

Nesta dissertação possui algumas limitações que se discutem de seguida. Uma das principais foi a indisponibilidade de conjuntos de dados específicos, o que direcionou à utilização de dados fictícios, como os gerados por modelos de linguagem como o ChatGPT. A ausência de dados reais pode ter impactado a capacidade de generalização dos resultados e a precisão das conclusões. Além disso, outra lacuna significativa foi a impossibilidade de testar o algoritmo desenvolvido num ambiente empresarial real. A validação prática do modelo em contexto real seria crucial para avaliar a eficácia e a escalabilidade da solução proposta. A falta de testes em ambiente empresarial limitou a capacidade de observar diretamente como a IA influencia na dinâmica das equipas e no desempenho dos projetos de *software* na prática.

Para ultrapassar as limitações identificadas durante a investigação, propõem-se o desenvolvimento de vários trabalhos futuros: utilizar um *dataset* real para os testes; encontrar uma empresa de desenvolvimento de software que esteja disponível para testar estes algoritmos e validar a eficácia dos mesmos e explorar mais técnicas avançadas de IA para além das utilizadas, como redes neuronais complexas. Por fim, investir em estudos de caso, a longo prazo independentemente do período para monitorizar o desempenho das equipas ao integrar IA na gestão de projetos de software nas empresa para compreender os efeitos causados na dinâmica das equipas.

## REFERÊNCIAS

- Ahmadi, A., Delkhosh, F., Deshpande, G., Patterson, R. A., & Ruhe, G. (2023). Learning *Software Project Management From Analyzing Q&A's in the Stack Exchange. IEEE Access, 11*, 5429–5441. <https://doi.org/10.1109/ACCESS.2023.3235953>
- Alan M. Turing. (n.d.). *Parsing the Turing Test Philosophical and Methodological Issues in the Quest for the Thinking Computer*.
- Allen, G., & Owens, M. (2010). Introducing SQLite. In *The Definitive Guide to SQLite* (pp. 1–16). Apress. [https://doi.org/10.1007/978-1-4302-3226-1\\_1](https://doi.org/10.1007/978-1-4302-3226-1_1)
- Alshaikhi, A., & Khayyat, M. (2021, March 30). An investigation into the impact of artificial intelligence on the future of project management. *2021 International Conference of Women in Data Science at Taif University, WiDSTaif 2021*. <https://doi.org/10.1109/WIDSTAIIF52235.2021.9430234>
- António Miguel. (2016). *Gestão Moderna de Projetos: Vol. 7ª Edição Atualizada* (Editora de Informática.Lda, Ed.; 7ª Edição Atualizada).
- Auth, G., Johnk, J., & Wiecha, D. A. (2021). A Conceptual Framework for Applying Artificial Intelligence in Project Management. *Proceedings - 2021 IEEE 23rd Conference on Business Informatics, CBI 2021 - Main Papers, 1*, 161–170. <https://doi.org/10.1109/CBI52690.2021.00027>
- Clinton, J., Khabaza, T., Reinartz, T., & Wirth, R. (1999). *The CRISP-DM Process Model*.
- Colomo-Palacios, R., González-Carrasco, I., López-Cuadrado, J. L., Trigo, A., & Varajao, J. E. (2014). I-Competere: Using applied intelligence in search of competency gaps in *software project managers. Information Systems Frontiers, 16*(4), 607–625. <https://doi.org/10.1007/s10796-012-9369-6>
- Dias, R., Sousa, F., & Trigo, A. (2023). Certified Accountants and ChatGPT. *XIX CONGRESSO INTERNACIONAL DE CONTABILIDADE E AUDITORIA*. <https://www.researchgate.net/publication/375162742>

*Composição de Equipas de Projetos de Desenvolvimento de Software com IA*

---

- Gomes, D. S. (2010). Inteligência Artificial: Conceitos e Aplicações. *Revista Olhar Científico-Faculdades Associadas de Ariquemes-V. 01, n.2, Ago./Dez.*
- Felipa Lopes dos Reis. (2018). *Investigação Científica e Trabalhos Académicos- Guia Prático* (Lda. Sílabo, Ed.; 1ª Impessão-Lisboa).
- Fernandes, G. L. (2019). *DESENVOLVIMENTO DE PROJETOS DE DATA SCIENCE*.  
<http://lattes.cnpq.br/0866467638692010>
- GSTEP©. (2020). <https://www.gstep.pt/>.
- <https://www.vosviewer.com/>. (2024).
- Kamthan, P., & Shahmir, N. (2019). On *software* projects in academia and industry from a perspective of *software* engineering education. *Proceedings - 6th Annual Conference on Computational Science and Computational Intelligence, CSCI 2019*, 34–39. <https://doi.org/10.1109/CSCI49370.2019.00013>
- Kanbur, M., Om Prakash, C., & Kulkarni, P. (2023). Creative AI in *Software* Project Management. *2023 2nd International Conference on Futuristic Technologies, INCOFT 2023*. <https://doi.org/10.1109/INCOFT60753.2023.10425234>
- Kitchenham, B. (2004). *Procedures for Performing Systematic Reviews*.
- Laureano, R. M. S., Caetano, N., & Cortez, P. (2014). Previsão de tempos de internamento num hospital português: Aplicação da metodologia CRISP-DM. *RISTI - Revista Iberica de Sistemas e Tecnologias de Informacao*, 13, 83–98.  
<https://doi.org/10.4304/risti.13.83-98>
- Lima, T., & Porto, J. B. (2019). *Análise de Soft Skills na Visão de Profissionais da Engenharia de Software*.
- Liubchenko, V. (2022). Specific Aspects of *Software* Development Process for AI/ML-based Systems. *International Scientific and Technical Conference on Computer Sciences and Information Technologies, 2022-November*, 470–473.  
<https://doi.org/10.1109/CSIT56902.2022.10000821>

*Composição de Equipas de Projetos de Desenvolvimento de Software com IA*

---

- Ma, Z., Li, R., Li, T., Zhu, R., Jiang, R., Yang, J., Tang, M., & Zheng, M. (2020). A data-driven risk measurement model of *software* developer turnover. *Soft Computing*, 24(2), 825–842. <https://doi.org/10.1007/s00500-019-04540-z>
- Mamatha, R., & Suma, K. G. (2021). Role of machine learning in *software* project management. *Journal of Physics: Conference Series*, 2040(1). <https://doi.org/10.1088/1742-6596/2040/1/012038>
- Mauricio Piacentini. (2024). *DB Browser for SQLite (DB4S)*. <https://sqlitebrowser.org/>
- Mönnich, A., Ronacher, A., Lord, D., Li, G., Bronson, J., Unterwaditzer, M., & Jones, P. (2024). *Flask*. <https://Flask.palletsprojects.com/en/3.0.x/>
- Mushtaq, Z., Alshmrany, S., Alturise, F., & Alkhalifah, T. (2022). Early Size and Effort Estimation of Mobile Application Development. *EAI Endorsed Transactions on Energy Web*, 9(37), 1–8. <https://doi.org/10.4108/eai.31-5-2021.170010>
- Nadkarni, P. M., Ohno-Machado, L., & Chapman, W. W. (2011). Natural language processing: An introduction. In *Journal of the American Medical Informatics Association* (Vol. 18, Issue 5, pp. 544–551). <https://doi.org/10.1136/amiajnl-2011-000464>
- Nethravathi, R., Kumar, S. N., Shwetha, S., Shyamsunder, M., & Reddy, C. V. K. (2020). A Redesigning *Software* Procedure in Improved *Software* Management using Machine Learning. *IOP Conference Series: Materials Science and Engineering*, 981(2). <https://doi.org/10.1088/1757-899X/981/2/022046>
- Niranjan, S. K., Institute of Electrical and Electronics Engineers. Bangalore Section, & Institute of Electrical and Electronics Engineers. (2016). *Proceedings of the 2016 2nd International Conference on Applied and Theoretical Computing and Communication Technology (iCATccT) : 21-23 July 2016, Bengaluru, Karnataka, India*.
- Nitin Rajadhyaksha, C., & Saini, J. R. (2022). Robotic Process Automation for *Software* Project Management. *2022 IEEE 7th International Conference for Convergence in Technology, I2CT 2022*. <https://doi.org/10.1109/I2CT54291.2022.9823972>

*Composição de Equipas de Projetos de Desenvolvimento de Software com IA*

---

- Pimenta, D., Teles, M., Belfo, F., & Trigo, A. (2023). Medication recommendation in cancer treatment based on cell line similarity. *Procedia Computer Science*, 219, 1493–1500. <https://doi.org/10.1016/j.procs.2023.01.440>
- PMI. (2017). *PMBOK® Guide – Sixth Edition* (Sixth Edit). Project Management Institute.
- PMI. (2019). *Pulse of the Profession 2019*.
- PwC. (2019). *22nd CEO Survey - Middle East Findings*. <https://www.pwc.com/m1/en/ceo-survey/2019.html>
- Radaideh, M. A. (2021). Shifting the paradigms from teaching project management to teaching *software* project management at Jordan university of science and technology based on the IEEE *software* engineering management knowledge area. *Proceedings - 2021 International Conference on Computational Science and Computational Intelligence, CSCCI 2021*, 1072–1078. <https://doi.org/10.1109/CSCCI54926.2021.00227>
- Riaz, M. Q., Abbas, M., & Irshad, S. (2017). Evaluation of Team Expertise in *Software Development Organizations*. *Proceedings - 2017 International Conference on Frontiers of Information Technology, FIT 2017*, 2017-January, 58–63. <https://doi.org/10.1109/FIT.2017.00018>
- Romero Izurieta, R., Toapanta Toapanta, S. M., Caucha Morales, L. J., Hifóng, M. M. B., Gómez Díaz, E. Z., Mafla Gallegos, L. E., Maciel Arellano, M. R., & Orizaga Trejo, J. A. (2022). Model to Optimize the Management of Strategic Projects Using Genetic Algorithms in a Public Organization. *Information (Switzerland)*, 13(11). <https://doi.org/10.3390/info13110533>
- Sarwar, B., Karypis, G., Konstan, J., & Riedl, J. (2001). Item-based collaborative filtering recommendation algorithms. *Proceedings of the 10th International Conference on World Wide Web*, 285–295. <https://doi.org/10.1145/371920.372071>
- Shafqat, M. A., Das, T., Ali, S. S., Raazi, S. M. K. U. R., Ali, F., & Mehmood, A. (2021, July 15). Challenges faced by Business Analysts eliciting User Requirements in the *software* industry: A survey-based study. *Proceedings of the 2021 Mohammad Ali*

*Composição de Equipas de Projetos de Desenvolvimento de Software com IA*

---

- Jinnah University International Conference on Computing, MAJICC 2021.*  
<https://doi.org/10.1109/MAJICC53071.2021.9526267>
- sigmoidal.ai. (2024). <https://sigmoidal.ai/alem-da-siri-como-a-ia-do-iphone-pode-ser-substituida-nos-proximos-anos/>.
- Smith, C. , M. B. , H. T. , & Y. (2006). *The History of Artificial Intelligence*.
- Taboada, I., Daneshpajouh, A., Toledo, N., & de Vass, T. (2023a). Artificial Intelligence Enabled Project Management: A Systematic Literature Review. *Applied Sciences (Switzerland)*, 13(8). <https://doi.org/10.3390/app13085014>
- Taboada, I., Daneshpajouh, A., Toledo, N., & de Vass, T. (2023b). Artificial Intelligence Enabled Project Management: A Systematic Literature Review. *Applied Sciences (Switzerland)*, 13(8). <https://doi.org/10.3390/app13085014>
- Varajão, J., Marques, R. P., & Trigo, A. (2022). Project Management Processes – Impact on the Success of Information Systems Projects. *Informatica*, 421–436. <https://doi.org/10.15388/22-INFOR488>
- Varajão, J., & Trigo, A. (2016). Evaluation of is project success in infsysmakers: An exploratory case study. *2016 International Conference on Information Systems, ICIS 2016*.
- Varajão, J., Trigo, A., Pereira, J. L., & Moura, I. (2022). Information systems project management success. *International Journal of Information Systems and Project Management*, 9(4), 62–74. <https://doi.org/10.12821/ijispm090404>
- Venkata Ramana, B., & Narsimha, G. (2022). Identification of the Ideal Team Capabilities and Predictive Success Measure for *Software* Projects Using Machine Learning. In *Lecture Notes in Networks and Systems* (Vol. 385). [https://doi.org/10.1007/978-981-16-8987-1\\_64](https://doi.org/10.1007/978-981-16-8987-1_64)
- Wang, Y., Pan, Y., Yan, M., Su, Z., & Luan, T. H. (2023). A Survey on ChatGPT: AI-Generated Contents, Challenges, and Solutions. *IEEE Open Journal of the Computer Society*, 4, 280–302. <https://doi.org/10.1109/OJCS.2023.3300321>

*Composição de Equipas de Projetos de Desenvolvimento de Software com IA*

---

Wirth, R., & Hipp, J. (2000). CRISP-DM: towards a standard process model for data mining. *Proceedings of the Fourth International Conference on the Practical Application of Knowledge Discovery and Data Mining, 1*, 29–40.

Younisse, R., & Azzeh, M. (2023). Application of Natural Language Processing Techniques in Agile Software Project Management: A Survey. *2023 14th International Conference on Information and Communication Systems, ICICS 2023*.  
<https://doi.org/10.1109/ICICS60529.2023.10330468>

## APÊNDICES

*Composição de Equipas de Projetos de Desenvolvimento de Software com IA*

**APÊNDICE 1. Conjunto de dados Projeto**

Nome	Descrição Projeto	Data_inicio	Data_fim	Equipas	Índice do Sucesso do Projeto
Migração de Sistema ERP para Nuvem	Transferência de um sistema de ERP (Enterprise Resource Planning) para uma infraestrutura de nuvem, visando escalabilidade e redução de custos operacionais.	2024-08-01	2025-03-15	Equipe A: Desenvolvimento Software	90%
Software de Simulação de Processo Industrial	Criação de um software para simulação de processos industriais, incluindo modelagem 3D e análise de desempenho.	2024-09-10	2025-04-30	Equipe A: Desenvolvimento de Software	80%
Desenvolvimento de Plataforma E-commerce	Criação de uma plataforma de comércio eletrônico personalizada, integrado de recursos avançados de segurança e análise de dados.	2024-11-01	2025-06-30	Equipe A: Desenvolvimento Web	70%
Campanha de Marketing Digital	Implementação de uma campanha de marketing digital para lançamento de uma aplicação	2024-09-10	2025-01-15	Equipa B: Marketing Digital	85%
Lançamento de Novo Produto	Planeamento e execução do lançamento de um novo smartphone com tecnologia inovadora.	2024-08-15	2025-02-28	Equipa B: Marketing Digital	50%
Sistema de Gestão Escolar	Desenvolvimento de um sistema integrado para gestão administrativa e académica de uma escola.	2024-07-01	2024-12-02	C: Gestores	40%
Aplicação de Gestão de Tarefas	Criação de uma aplicação para gestão de tarefas pessoais e profissionais, com funcionalidades avançadas de colaboração e sincronização em tempo real.	2024-07-15	2025-01-31	C: Gestores	30%
Desenvolvimento de Aplicativo de Gestão de Tarefas	Criação de um aplicativo para gerenciamento de tarefas pessoais e profissionais, com funcionalidades avançadas	2024-07-15	2025-01-31	Equipa D: Consultoria Externa de UX/UI Design	80%

*Composição de Equipas de Projetos de Desenvolvimento de Software com IA*

	de colaboração e sincronização em tempo real.				
Desenvolvimento de Plataforma de E-commerce	Criação de uma plataforma de comércio eletrónico personalizada, integrando recursos avançados de segurança e análise de dados.	2024-11-01	2025-06-30	Equipa D: Consultoria Externa de UX/UI Design	
Implementação de Sistema de Gestão Financeira	Desenvolvimento e implementação de um sistema de gestão financeira integrado para uma empresa, incluindo contabilidade, folha de pagamento e relatórios fiscais.	2024-07-01	2025-01-31	Equipe E: Contabilista e Consultoria Fiscal	
Sistema de Gestão de Orçamento Empresarial	Desenvolvimento de um sistema para gestão de orçamento empresarial, que inclui planeamento, controle de custos e relatórios financeiros.	2024-09-01	2025-05-31	Equipe E: Contabilista e Consultoria Fiscal	

*Composição de Equipas de Projetos de Desenvolvimento de Software com IA*

**APÊNDICE 2. Conjunto de dados Colaboradores**

Funcionários	Descrição e nível de competências
João Silva	Desenvolvimento de aplicações móveis: 4 Integração com APIs externas: 3 Segurança de dados: 4
Maria Santos	Desenvolvimento web: 5 Base de dados: 4 Algoritmos de otimização: 3
Pedro Almeida	Desenvolvimento de interfaces: 4 Integração de sistemas: 4 Desenvolvimento de aplicações móveis: 3
Ana Costa	Desenvolvimento web: 3 Integração com APIs externas: 2 Segurança de dados: 3
Tiago Pereira	Desenvolvimento de aplicações móveis: 3 Base de dados: 3 Algoritmos de otimização: 2
Inês Oliveira	Desenvolvimento web: 4 Integração de sistemas: 4 Desenvolvimento de interfaces: 3
Miguel Sousa	Desenvolvimento de aplicações móveis: 2 Integração com APIs externas: 3 Segurança de dados: 2
Sofia Fernandes	Desenvolvimento web: 5 Base de dados: 4 Algoritmos de otimização: 4
André Martins	Desenvolvimento de interfaces: 3 Integração de sistemas: 3 Desenvolvimento de aplicações móveis: 2
Carolina Ferreira	Desenvolvimento web: 4 Integração com APIs externas: 4 Segurança de dados: 3
Ricardo Mendes	Desenvolvimento de aplicações móveis: 3 Base de dados: 3 Algoritmos de otimização: 3
Diana Ribeiro	Desenvolvimento web: 3 Integração de sistemas: 2 Desenvolvimento de interfaces: 2
Bruno Carvalho	Desenvolvimento de aplicações móveis: 4 Integração com APIs externas: 3 Segurança de dados: 4
Catarina Lopes	Desenvolvimento web: 4 Base de dados: 4 Algoritmos de otimização: 3
Hugo Rodrigues	Desenvolvimento de interfaces: 3 Integração de sistemas: 3 Desenvolvimento de a aplicações móveis: 3
Marta Gonçalves	Desenvolvimento web: 5 Integração com APIs externas: 4 Segurança de dados: 4
Rui Correia	Desenvolvimento de aplicações móveis: 3 Base de dados: 3 Algoritmos de otimização: 3
Margarida Castro	Desenvolvimento web: 3 Integração de sistemas: 2

*Composição de Equipas de Projetos de Desenvolvimento de Software com IA*

---

	Desenvolvimento de interfaces: 2
Francisco Ramos	Desenvolvimento de aplicações móveis: 4 Integração com APIs externas: 3 Segurança de dados: 4
Beatriz Moreira	Desenvolvimento web: 4 Base de dados: 4 Algoritmos de otimização: 3

*Composição de Equipas de Projetos de Desenvolvimento de Software com IA*

---

## **ANEXOS**

## ANEXO 1. Caso de Sucesso 1 da Empresa GESTP

**GESTP**  
business intelligence

CASO de  
**SUCESSO**

GESTP automatiza solução de Planeamento e Controlo Orçamental numa das maiores Seguradoras presentes em Portugal

Utilizando a ferramenta Oracle Hyperion, o projeto de Planeamento e Controlo de Gestão (PCG) desenvolvido pela GESTP teve como objetivo reduzir o esforço no tratamento de dados financeiros da Companhia de Seguros, produzir relatórios para controlo orçamental e aumentar o tempo para analisar resultados, indicadores e tendências.

**Desafio**

A Companhia de Seguros internacional, implantada em Portugal e líder no seu setor de mercado, fazia todo o planeamento e controlo orçamental através de folhas de cálculo Excel que, numa empresa desta dimensão, levantava os seguintes problemas:

- Esforço de cerca de 80% no tratamento de dados, tendo apenas 20% para análise dos mesmos
- Dificuldade na criação de diversos cenários de orçamento
- Gestão de versões de ficheiros de Excel
- Informação centralizada no departamento de Planeamento e Controlo de Gestão, sendo que sempre que alguma área de negócio precisava de analisar os desvios tinha de esperar para receber essa informação tratada
- Risco de perder informação caso algum ficheiro fosse danificado
- Risco de falhas de segurança

Por forma a reduzir o esforço no tratamento de dados e aumentar o tempo para analisar resultados, indicadores e tendências, a Seguradora optou por automatizar o processo de forma a colocar todo o know-how num software user-friendly. Pretendia-se assim atingir maior autonomia e interatividade por parte dos utilizadores e fazer um controlo de inputs, versões e acessos mais eficiente.

De acordo com as necessidades apresentadas e com o apoio da gestão de topo, a Seguradora selecionou a GESTP devido à sua experiência e know-how na área de EPM (Enterprise Performance Management) e à relação de confiança gerada ao longo de vários projetos realizados em conjunto.

**Resolução**

A informação disponível no mundo é cada vez maior e mais acessível, todos estamos cada vez mais informados e naturalmente quem souber interpretar melhor essa crescente informação terá vantagem neste mercado cada vez mais competitivo. Nesse sentido, o aumento do nível de detalhe, a integração de outras/novas variáveis e a criação de cenários serão temas que farão certamente parte de futuros desenvolvimentos com a GESTP.

Responsável de Planeamento e Gestão da Informação da Seguradora

CASO SUCESSO

## ANEXO 2. Caso de Sucesso 2 da Empresa GESTP



**GSTEP**  
business intelligence

CASO de  
**SUCESSO**

GSTEP agiliza negócio da Lusitania Seguros

Para gerir os crescentes requisitos de negócio da Lusitania, que exigem respostas em tempo real, a GSTEP implementou o Oracle Exalytics, a primeira máquina de BI (Business Intelligence) in-memory do setor que integra Hardware e Software, proporcionando elevada capacidade de resposta, redução do custo e da complexidade da infraestrutura de TI, e aumento da produtividade e desempenho. O projeto permitiu ainda melhorar a rapidez e a usabilidade da ferramenta de suporte à decisão da Lusitania.

**“** Além das competências técnicas que detém e domina, a GSTEP transmite-nos confiança. Temos plena confiança no que a equipa nos sugere e no que desenvolve em parceria connosco. Dizer que (a confiança) é a chave do sucesso do nosso relacionamento. Além disso, a GSTEP é Gold Partner da Oracle e detém um elevado nível de certificação e experiência nas novas áreas de crescimento do Business Intelligence (BI), factos que evidenciam a sua competência para este projeto. **”**

**Alexandre Ramos**  
CEO  
Lusitania Seguros

**LUSITANIA**  
Seguros

**Desafio**

Como uma das maiores Seguradoras a operar em Portugal, a Lusitania enfrentava desafios de rapidez de resposta às suas necessidades de negócio.

A companhia de Seguros utilizava uma tecnologia em descontinuação, decidindo avançar com uma prova de conceito de forma a selecionar a tecnologia e o parceiro implementador que melhor respondiam às suas necessidades atuais e futuras, entre as quais, aumentar a performance e a usabilidade da ferramenta de suporte à decisão.

**Solução**

Para responder aos objetivos apresentados pela Lusitania, a GSTEP implementou o Oracle Exalytics, a primeira máquina BI in-memory do setor, que fornece o desempenho mais rápido para aplicações de Business Intelligence e de planeamento. Esta solução integra Hardware e Software de forma a tirar o melhor proveito de ambos, proporcionando elevada capacidade de resposta, redução do custo e da complexidade da infraestrutura de TI, e aumento da produtividade e desempenho.

O Oracle Exalytics foi desenhado para fornecer às organizações dados orientados à vantagem competitiva, sendo uma solução para gerir os crescentes requisitos de negócio, que exigem respostas em tempo real.

Com cerca de 1.000 utilizadores, a solução tem diversas possibilidades em termos de formatos de interação pelo utilizador final, nomeadamente web tradicional, Mobile, Desktop e MS Office.

caso de sucesso 01