

Automação de Integrações em Marketing Omnicanal

FRANCISCO JOSÉ BORGES BOBIANO

outubro de 2021

Automação de Integrações em Marketing Omnicanal

Francisco José Borges Bobiano

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

**Orientador: Prof. Fernando Jorge Duarte
Supervisor: Ivo Pereira**

Dedicatória

Aos meus pais por estarem sempre presentes.

Resumo

A presente dissertação pretende descrever o processo de investigação e desenvolvimento do projeto realizado no âmbito da unidade curricular de Tese/Dissertação/Estágio do Mestrado em Engenharia Informática, na área de especialização de Engenharia de Software, no Instituto Superior de Engenharia do Porto.

O projeto, proposto pela empresa E-goi, apresenta o objetivo principal de investigar e desenvolver uma ferramenta robusta e escalável que permita a gestão de *workflows* de integração, baseados em *triggers* e *actions* de modo a alargar as possibilidades de integrações com o E-goi.

O cliente moderno procura sobretudo serviços mais *ready to go* onde não há dependência de perfis como designer ou programador. Com base nesta premissa a importância das integrações e a sua automação tem vindo a aumentar, existindo uma necessidade de agilizar o processo de integração de diversas plataformas.

A descrição técnica deste documento descreve o processo de desenvolvimento da solução, resultante da análise do problema e levantamento dos requisitos de *software*.

Palavras-chave: *Marketing* Digital, Automação de *workflows*, Integrações

Abstract

This dissertation aims to describe the process of research and development of the project carried out within the Thesis/Dissertation/Internship of the Master's degree in Computer Engineering, in the area of software engineering specialization, at the Instituto Superior de Engenharia do Porto.

The project, proposed by the company E-goi, presents the main objective of investigating and develop a robust and scalable tool that allows the management of integration workflows, wich are trigger and action based to extend the possibilities of integrations with E-goi.

The modern client mainly seeks more ready to go services where there is no dependence on profiles as a designer or programmer. Based on this premise, the importance of integrations and their automation has been increasing, and there is a need to streamline the integration process of various platforms.

The technical description of this document describes the process of developing the solution, resulting from the analysis of the problem and survey of software requirements.

Agradecimentos

Em primeiro lugar, agradeço ao professor Fernando Jorge Duarte pelo acompanhamento feito na elaboração deste documento.

Agradeço à E-goi pela oportunidade de realizar este projeto e pelas excelentes condições de trabalho proporcionadas.

Um agradecimento especial a todos os amigos que me acompanharam, ajudaram e viveram esta etapa comigo. Estarei sempre grato.

Por último, o maior dos agradecimentos aos meus pais e à minha família, a vossa presença em todas as etapas da minha vida tornaram isto possível. Um muito obrigado.

Conteúdo

Lista de Figuras	xv
Lista de Tabelas	xvii
Lista de Acrónimos	xxi
Glossário	xxiii
1 Introdução	1
1.1 Apresentação da Empresa	1
1.2 Descrição do Problema	1
1.3 Objetivos	2
1.4 Estrutura do Documento	3
2 Estado da Arte	5
2.1 Contextualização	5
2.1.1 Lista de Contactos	5
2.1.2 Tags	5
2.1.3 Trigger e Action	6
2.1.4 Workflow	6
2.2 Análise do E-goi	6
2.3 Automação de Workflows	7
2.4 Soluções existentes	8
2.4.1 Zapier	8
2.4.2 Microsoft Power Automate	10
2.4.3 Integromat	11
2.4.4 IFTTT	13
2.4.5 Automate.io	14
2.4.6 n8n.cloud	15
2.4.7 Workato	17
2.4.8 Análise Comparativa	18
3 Análise	23
3.1 Engenharia de Requisitos	23
3.1.1 Requisitos Funcionais	23
UC01 - Adicionar um Node ao Workflow	24
UC02 - Criar Credenciais	24
UC03 - Configurar Node E-goi	25
UC04 - Conectar Nodes	26
UC05 - Ativar Workflow	27
UC06 - Ver Execuções	28

UC07 - Ver Workflows	28
3.1.2 Requisitos Não Funcionais	29
3.2 Análise de Valor	30
3.2.1 Processo de Inovação	30
3.2.2 New Concept Development	31
Identificação da Oportunidade	32
Análise da Oportunidade	32
Geração de Ideias	35
Seleção de Ideias	36
Definição do Conceito	39
3.2.3 Valor da Solução	39
Valor, Valor para o Cliente, Valor Percepcionado	39
Proposta de Valor	40
3.2.4 Business Model Canvas	40
3.3 Sumário	42
4 Desenho	43
4.1 Desenho Arquitetural	43
4.1.1 Domínio	44
4.1.2 Definição das alternativas	45
4.1.3 Vista de Implementação	45
Alternativa 1	45
Alternativa 2	46
4.1.4 Vista de Implantação	47
Alternativa 1	47
Alternativa 2	48
4.1.5 Modelo de Dados	48
4.1.6 Escolha das Alternativas	49
4.1.7 Vista de Processos	50
Configurar Node E-goi	50
4.2 Sumário	51
5 Implementação	53
5.1 Metodologia de Criação de Novos Nodes	53
5.2 Criação Node E-goi	54
5.2.1 Configurar Credenciais	58
5.2.2 Publicação do node	59
5.3 Interface Gráfica	59
5.4 Instâncias	61
5.5 Base de Dados	61
5.6 Sumário	62
6 Experimentação e Avaliação	63
6.1 Métricas	63
6.2 Hipóteses	63
6.3 Metodologia	64
6.4 Análise de Resultados	64
6.5 Sumário	66
7 Conclusão	67

7.1	Objetivos concluídos	67
7.2	Limitações	68
7.3	Trabalho Futuro	69
Bibliografia		71
A Método AHP		75
A.1	Escala Fundamental de Saaty	75
A.2	Valores de IR para matrizes quadradas de ordem n	75
A.3	Estabelecer Prioridades Relativas	76
A.4	Avaliar Consistência das Prioridades Relativas	76
A.5	Construir Matriz da Comparação Paritária	77
B Código de implementação		79
B.1	Ficheiro GenericFunctions	79
B.2	Método execute	80
B.3	Ficheiro Interfaces	80
B.4	Ficheiro Docker-compose	81
B.5	Ficheiro .env	82
C Inquérito de Satisfação		83
D Respostas do Inquérito de Satisfação		87

Lista de Figuras

2.1	Homepage do Goidini	7
2.2	Exemplo de fluxo na plataforma Zapier	9
2.3	Planos da plataforma Zapier	10
2.4	Exemplo de fluxo na plataforma Microsoft Power Automate	11
2.5	Planos da plataforma Microsoft Power Automate	11
2.6	Exemplo de fluxo na plataforma Integromat. Fonte: [14]	12
2.7	Planos da plataforma Integromat	13
2.8	Planos da plataforma IFTTT	14
2.9	Exemplo de fluxo na plataforma Automate.io. Fonte: [17]	15
2.10	Planos da plataforma Automate.io	15
2.11	Exemplo de fluxo na plataforma n8n.cloud	16
2.12	Planos da plataforma n8n.cloud	17
2.13	Exemplo de fluxo na plataforma Workato. Fonte: [21]	18
3.1	Diagrama de Casos de Uso	23
3.2	SSD do UC01	24
3.3	SSD do UC02	25
3.4	SSD do UC03	26
3.5	SSD do UC04	27
3.6	SSD do UC05	28
3.7	SSD do UC06	28
3.8	SSD do UC07	29
3.9	Modelo de processo de valor. Fonte: [30]	31
3.10	Modelo de New Concept Development. Fonte: [30]	32
3.11	Análise SWOT	33
3.12	Gráfico de pesquisas sobre " <i>Workflow Automation</i> " entre 2016 e 2021	34
3.13	Estrutura Hierárquica do Analytic Hierarchy Process (AHP)	36
3.14	Modelo Canvas	41
4.1	Modelo de Domínio	44
4.2	Alternativa 1 de Diagrama de Componentes	45
4.3	Alternativa 2 de Diagrama de Componentes	46
4.4	Alternativa 1 de Diagrama de Implantação	47
4.5	Alternativa 2 de Diagrama de Implantação	48
4.6	Modelo de dados	48
4.7	Diagrama de sequência do UC03	50
5.1	Diagrama de Pacotes	53
5.2	Publicação do node E-goi	59
5.3	Área de criação de Workflows	60
5.4	Área de configuração do node E-goi	60

A.1	Tabela de valores IR para matrizes quadradas de ordem n .	75
C.1	Início do inquérito	83
C.2	Início da secção de utilização geral da plataforma	84
C.3	Fim da secção de utilização geral da plataforma	85
C.4	Secção de criação de workflows com o node E-goi	86
D.1	Gráfico de barras das perguntas 1 e 2 da primeira secção	87
D.2	Gráfico de barras das perguntas 3 e 4 da primeira secção	88
D.3	Gráfico de barras das perguntas 5 e 6 da primeira secção	89
D.4	Gráfico de barras das perguntas 1 e 2 da segunda secção	90
D.5	Gráfico de barras das perguntas 3 e 4 da segunda secção	91
D.6	Gráfico de barras da pergunta 5 da segunda secção	91

Lista de Tabelas

2.1	Comparação entre as diversas plataformas	20
3.1	Tabela FURPS+	30
3.2	Matriz de comparação dos critérios do segundo nível	37
3.3	Matriz de comparação paritária de recursos tecnológicos	38
3.4	Matriz de comparação paritária de facilidade de integração	38
3.5	Matriz de comparação paritária de tempo de desenvolvimento	38
3.6	Matriz de prioridades relativas das abordagens	39
3.7	Matriz de prioridades relativas dos critérios	39
3.8	Matriz de prioridade composta	39
6.1	Escala de classificação das questões	64
6.2	Resultados do inquérito de satisfação da utilização da plataforma	65
6.3	Resultados do inquérito de satisfação da utilização da plataforma	65
A.1	Escala fundamental de Saaty	75
A.2	Matriz normalizada dos critérios do segundo nível	76
A.3	Matriz normalizada de recursos tecnológicos do segundo nível	77
A.4	Matriz normalizada de facilidade de integração do segundo nível	77
A.5	Matriz normalizada de tempo de desenvolvimento do segundo nível	77

Lista de Código

5.1	Parâmetro resource	55
5.2	Parâmetro operation	56
5.3	Exemplo de loadOptionsMethod	57
5.4	Exemplo de displayOptions	57
5.5	Ficheiro EgoiApi.credentials.ts	58
5.6	Ficheiro EgoiApi.credentials.ts	58
5.7	Exemplo da definição das tabelas	62
B.1	função EgoiApiRequest	79
B.2	Parte do método execute	80
B.3	Ficheiro Interfaces	80
B.4	Exemplo de implementação do ficheiro docker-compose.yml	81
B.5	Exemplo do ficheiro .env	82

Lista de Acrónimos

AHP	Analytic Hierarchy Process.
API	Interfaces de Programação de Aplicação.
HTTPS	Hyper Text Transfer Protocol Secure.
JSON	JavaScript Object Notation.
NCD	New Concept Development.
NPD	New Product Development.
PHP	Hypertext Preprocessor.
REST	Representational State Transfer.
SQL	Structured Query Language.

Glossário

Drag-and-drop	Ação de clicar num objeto virtual e arrastá-lo para uma posição diferente.
Fair-code	Licença que permite ver o source code e fazer software gratuito.
Git	Sistema de controlo de versões.
GitHub	Plataforma de hospedagem de código fonte e arquivos.
MariaDB	Sistema de gestão de bases de dados relacional com base em MySQL.
Monorepo	Estratégia de desenvolvimento de software onde o código de vários projetos é armazenado no mesmo repositório.
MySQL	Sistema de gestão de base de dados, que utiliza a linguagem SQL como interface.
OAuth2	Protocolo de autorização aberta.
SaaS	System as a Service.
TypeScript	Linguagem de programação.
Webhook	Callback HTTP associado a uma ação..
Workflow	Fluxo de trabalho.

Capítulo 1

Introdução

O mercado atual é cada vez mais digital, as empresas e organizações necessitam de se diferenciar e inovar, tornando o marketing uma atividade chave. A maioria das transações online acontecem sem interação humana, sendo que as organizações procuram serviços mais *ready to go*, onde não há dependência de perfis como o designer ou programador. Automatizar processos tornou-se uma necessidade de forma a reduzir custos e tempos de execução[1].

1.1 Apresentação da Empresa

A E-goi é uma empresa especializada em *marketing*, fundada por Miguel Gonçalves em 2008 e está sediada em Matosinhos, Porto. A sua missão é “Criar soluções de comunicação digital, acessíveis a todos os profissionais de marketing do mundo. Torná-las simples e divertidas de usar, desafiando-nos todos os dias.” [2]. A E-goi conta com mais de 120 colaboradores e continua o seu processo de crescimento, sendo que em dezembro de 2020, o seu produto conta com mais de seiscentas e cinquenta mil contas criadas em mais de quarenta países.

O seu produto, o E-goi, consiste num SaaS (*Software as a Service*) de automação de marketing multicanal. Oferece funcionalidades desde criação e gestão de listas de contactos, criação e envio de campanhas por email e SMS, criação de automatizações, criação e gestão de *landing pages*, entre outras. É possível integrar o E-goi com centenas de aplicações, tais como WordPress, WooCommerce, PrestaShop, Shopify, Magento, Facebook, entre outras. Todas as integrações possíveis estão presentes no Goidini [3], que é o diretório de integrações da E-goi.

1.2 Descrição do Problema

A plataforma de automação de marketing da E-goi envia milhares de campanhas todos os dias, repartidas pelos vários canais de comunicação, dos quais se destacam E-mail, SMS, SmartSMS, Push, WebPush, Post Social, entre outros.

A automação de marketing é o uso de software para automatizar processos de marketing, como gestão de campanhas, gestão de contactos, segmentação de contactos, entre outros. Esta automação pode ajudar na centralização de dados e na criação de uma estratégia de marketing mais complexa [4].

As integrações são parte fundamental num sistema, pois permitem unificar vários produtos de forma a facilitar as necessidades do seu utilizador. Formam-se complexos sistemas que são

compostos por diversos componentes que apresentam diversas tecnologias para satisfazer necessidades complexas e únicas [5].

Na E-goi existe um diretório de integrações denominado Goidini [3]. Nele estão presentes todas as integrações possíveis com a plataforma E-goi. No entanto, esse diretório não permite automatizar ações entre diversas aplicações, ou integrações existentes no E-goi.

O cliente moderno procura sobretudo serviços mais *ready-to-go* onde não há dependência de perfis como designer ou programador. Com base nesta premissa, a importância das integrações e a sua automatização tem vindo a aumentar, existindo a necessidade de agilizar o processo de integração de diversas plataformas com o E-goi. Deste modo, o projeto a ser desenvolvido pretende responder a estas problemáticas e alargar as possibilidades de integrações com o E-goi.

Atualmente a E-goi apresenta uma integração com a plataforma Zapier (que será aprofundada no Capítulo 2). Esta plataforma oferece a possibilidade de automação de *workflows*. No entanto, o plano gratuito desta plataforma é bastante limitador relativamente às execuções de fluxos por mês e à criação de fluxos mais complexos.

Em suma, os problemas relativos à automação de integrações de marketing omnicanal da plataforma E-goi são:

- Existência de um ambiente limitado que automatize *workflows*;
- Estratégias de marketing cada vez mais complexas;
- Demanda de serviços mais *ready-to-go*.

A identificação de problemas deve ser desenvolvida cuidadosamente e é de elevada importância que seja feita com o maior rigor possível.

1.3 Objetivos

O principal objetivo do projeto é o desenvolvimento de uma ferramenta robusta e escalável que permita a gestão de *workflows* de integração, baseados em *triggers* e *actions* de modo a alargar as possibilidades de integrações com o E-goi.

É necessário definir objetivos específicos que correspondem aos resultados concretos que o projeto pode alcançar e que contribuem para a concretização do objetivo principal. Esta desconstrução dos objetivos, permite um melhor entendimento do projeto e dos resultados esperados, servindo como um guia de conteúdo do que será abordado. Assim, foram definidos os seguintes objetivos:

- Identificar problemas relacionados com as integrações na E-goi. Para tal, é necessário estudar o ambiente de integrações presente na plataforma e o fluxo que é geralmente usado nas integrações;
- Análise do fluxo geralmente utilizado nas integrações. Para completar este objetivo, é essencial que sejam estudadas diversas integrações existentes de forma a ter um melhor entendimento do processo;
- Identificar soluções para os problemas identificados. É importante que sejam estudadas diversas soluções para responder aos problemas identificados;

- Definir uma arquitetura para a melhor solução encontrada. Após identificar a melhor solução para colmatar os problemas identificados, é necessário definir as componentes do projeto e os moldes para a sua implementação;
- Implementar a arquitetura definida. Fase de desenvolvimento da solução com base nas boas práticas de engenharia informática;
- Integrar a solução desenvolvida na plataforma E-goi;
- Testar a solução. Por fim, após o desenvolvimento e integração da solução, é necessário testar de forma a perceber erros no seu desenvolvimento e verificar se contempla todos os objetivos propostos.

1.4 Estrutura do Documento

Este documento é composto por seis capítulos, sendo eles Introdução, Estado de Arte, Análise de Valor, Design da Solução, e Experimentação e Avaliação.

Introdução: capítulo subdividido em Apresentação da Empresa, Descrição do Problema, Objetivos e Estrutura do Documento. Tem como objetivo enquadrar o leitor no problema identificado.

Estado de Arte: neste capítulo é feita uma explicação de conceitos necessários para a total compreensão do documento, uma análise tecnológica à organização, um estudo de soluções e tecnologias semelhantes ao problema identificado e a sua comparação e análise.

Análise: capítulo que confirma a necessidade do projeto.

Desenho da Solução: capítulo onde são apresentados diagramas referentes à solução.

Experimentação e Avaliação: capítulo onde são apresentados os testes a serem implementados de forma a avaliar a solução.

Capítulo 2

Estado da Arte

Neste capítulo é apresentada uma análise à plataforma Goidini e é feito um estudo sobre diversas soluções existentes sobre gestão de workflows de integração. Por fim, é realizada uma análise comparativa, de forma a aprofundar o conhecimento existente na área de automação de integrações.

2.1 Contextualização

Esta secção tem o intuito de apresentar conceitos associados ao projeto descrito nesta dissertação, de forma a ter uma compreensão total do documento.

2.1.1 Lista de Contactos

Na plataforma E-goi, existem várias listas de contactos associadas a cada conta. Estas listas podem ser criadas ou importadas. As listas tem como objetivo permitir o envio automático de mensagens ou campanhas para diversos contactos.

Cada lista é composta por um conjunto de contactos, que por sua vez, são compostos por campos base e campos extras. Os campos base são o primeiro nome, último nome, telemóvel, data de nascimento, linguagem e email. O email é considerado um campo único pois identifica o contacto e diferencia-lo dos restantes. Os campos extra podem ser criados pelo cliente. A definição dos campos extra permite ao dono da lista uma liberdade de personalização e identificação de um contacto, conseguindo realizar campanhas mais personalizadas.

2.1.2 Tags

Dentro das listas de contactos é possível criar secções. A sua criação é possível utilizando *tags* personalizadas, definidas pelo utilizador e dono da lista de contactos. As *tags* são atribuídas a um contacto conforme as interações que este tem com o conteúdo selecionado e produzido pelo dono da lista de contactos. Por exemplo, se for enviada uma campanha de SMS com uma promoção e um link para aceder a essa promoção, é possível diferenciar quem abre o link e quem não abre o link. A diferenciação passa pela atribuição de uma *tag*. Deste modo é possível realizar uma campanha para os contactos que não abriram o link, de modo a aumentar a conversão do produto em questão.

2.1.3 Trigger e Action

No âmbito do projeto, *trigger* e *action* são componentes que representam uma aplicação ou serviço de integração. O *trigger* é o componente que ativa e começa um fluxo. As *actions* estão presentes no restante fluxo, sendo caracterizadas por permitirem a entrada de dados e o seu processamento.

2.1.4 Workflow

O *workflow* representa o fluxo constituído por um único *trigger* e uma ou mais ações. O fluxo é executado sempre que o *trigger* é ativado e termina quando todas as ações forem executadas.

2.2 Análise do E-goi

O E-goi é um produto bastante complexo, em constante desenvolvimento e evolução. A sua solução utiliza várias tecnologias, tais como PHP, Python, Angular, Javascript, HTML, CSS, entre outras.

Dependendo dos objetivos de cada componente, serão aplicadas as tecnologias mais apropriadas.

Goidini

O Goidini é uma plataforma desenvolvida para ser um ponto único de acesso para as diversas integrações que o E-goi apresenta com outras aplicações. Trata-se de um diretório de integrações e apresenta um sistema de login que está associado com o sistema de *login* do produto E-goi. Na sua página principal, Figura 2.1, apresenta as integrações disponíveis. É na página principal do Goidini que deverá estar acessível a solução a ser desenvolvida.

Esta unificação oferece aos clientes E-goi um ponto único para integrarem o E-goi com qualquer aplicação de uma forma simples.

O Goidini apresenta um sistema de gestão de utilizadores, que está relacionado com o sistema de gestão de utilizadores do E-goi. Um utilizador do Goidini tem de ser também um cliente E-goi. Este sistema permite guardar as integrações que um utilizador realizará.

Stack tecnológica do Goidini:

- Zend Framework v1.12.3 (PHP 5.6.40);
- Bootstrap V3;
- MySQL 1.38-MariaDB.

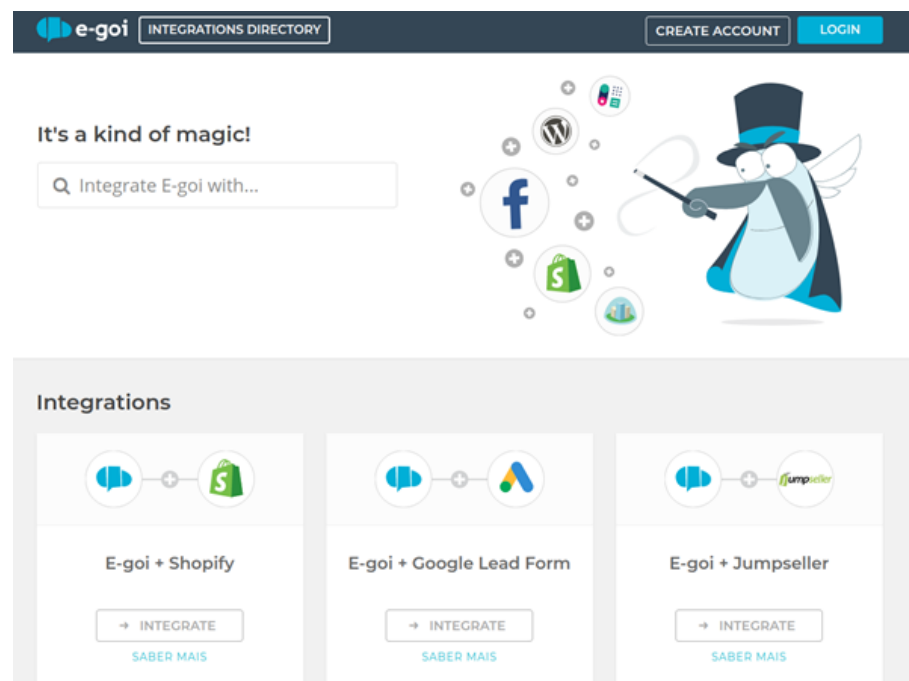


Figura 2.1: Homepage do Goidini

2.3 Automação de Workflows

Em todas as organizações há procedimentos que devem ser seguidos com o objetivo de proporcionar uma maior eficiência, consistência e qualidade. No entanto, o grau de concretização destes procedimentos é afetada pela performance da pessoa que executa as tarefas. Erros humanos, ansiedade e pressão de prazos de entrega são alguns dos aspectos que podem afetar a performance [6].

O conceito de automação de *workflows* incorpora a ideia de controlar e coordenar atividades e processos complexos. O objetivo alcançado é um trabalho computadorizado com grande qualidade e eficiência [6].

Darrel West afirma que aproveitando o extraordinário poder de processamento e armazenamento dos computadores, os humanos podem aumentar as suas habilidades e melhorar a sua produtividade [7].

É perceptível o potencial que existe em automatizar processos e *workflows*, não só por se realizar uma maior quantidade de tarefas, como se liberta recursos humanos para tarefas mais importantes dentro de uma organização. Quando combinado o processo de automização com a necessidade de transportar dados por diversas aplicações, é introduzido o conceito de integração.

A integração é o ato de juntar componentes menores num sistema que funciona como um. Refere-se ao resultado final de um processo que visa juntar subsistemas diferentes, muitas vezes díspares, para que os dados contidos em cada um se tornem parte de um sistema maior e abrangente que, idealmente, partilha dados de forma rápida e fácil [8].

Assim, automação de *workflows* de integração usa conectores Interfaces de Programação de Aplicação (API) para conectar diversas aplicações SaaS (*System as a Service*) e microserviços [9]. Apresenta o objetivo de automatizar processos complexos, conectando sistemas e transferindo dados fácil e eficazmente.

2.4 Soluções existentes

No mercado de automação de integrações existem diversas abordagens distintas, com vantagens e desvantagens associadas, sendo importante compreender em que pontos diferem as soluções existentes e quais são as suas principais vantagens.

Para contemplar estas necessidades, foram identificadas sete soluções que permitem a gestão de *workflows* e a sua automação. A análise destas, aborda os pontos positivos e negativos, as suas principais funcionalidades e a comparação de acordo com os parâmetros definidos.

2.4.1 Zapier

O Zapier [10] é uma plataforma de automação criada em 2011 por Wade Foster, Bryan Helming e Mike Knoop em Columbia, Missouri. Esta plataforma serve de ponte de conexão entre duas ou mais aplicações e serviços, para que executem determinadas ações de forma automática conforme determinado tipo de *trigger*.

No seu produto, define-se o *workflow* como *zap*, que é o ambiente que conecta as aplicações e serviços. Cada *zap* consiste num *trigger* e numa ou mais ações. Quando o *workflow* está ativo, todo o processo será executado sempre que o *event trigger* ocorra [11].

Relativamente às suas funcionalidades e características, o Zapier é reconhecido pela segurança dos dados dos *workflows* tendo uma equipa de segurança disponível vinte e quatro horas por semana que monitoriza notificações sobre software de terceiros. Oferece a possibilidade de realizar múltiplas tarefas com diferentes aplicações e ferramentas com um único *trigger*. Nomeiam-se quatro ações que acrescentam valor ao Zapier:

- Possibilidade de criação de diferentes caminhos pois eliminam a necessidade de criar vários *workflows* para satisfazer um fluxo;
- Filtragem de dados obtidos em ações prévias, sendo que o fluxo só prossegue se a condição for atendida;
- Formatação de dados obtidos em ações anteriores de forma a satisfazer as necessidades dos clientes;
- Temporizador para colocar em pausa o *workflow* durante um determinado período de tempo.

Apresenta mais de 2000 aplicações para a criação e automação de *workflows* e, juntamente com o seu design simples e intuitivo, são os principais pontos fortes desta plataforma. Como é uma das principais plataformas de automação no mercado e apresentar um leque vasto de aplicações, a E-go apresenta uma integração disponível nesta plataforma. Nesta integração estão disponíveis dois *triggers* e sete ações:

Triggers:

- Um novo subscritor é adicionado a uma lista de contactos;

- Um subscritor é removido de uma determinada lista de contactos.

Ações:

- Adição ou atualização de um contacto a uma determinada lista de contactos;
- Remoção de um contacto a uma lista de contactos;
- Adição de uma tag a um contacto existente;
- Envio de uma campanha de e-mail a um contacto;
- Envio de uma campanha de SMS a um contacto;
- Envio de uma SMS a um determinado contacto;
- Inicialização de um *autobot*.

A Figura 2.2 representa um exemplo de um fluxo com a integração da E-goi onde o fluxo inicia-se sempre que existe um novo cliente numa loja da plataforma WooCommerce. As informações são utilizadas para criar um novo contacto numa lista de contactos do E-goi, enviado esse contacto para uma Google Sheets.

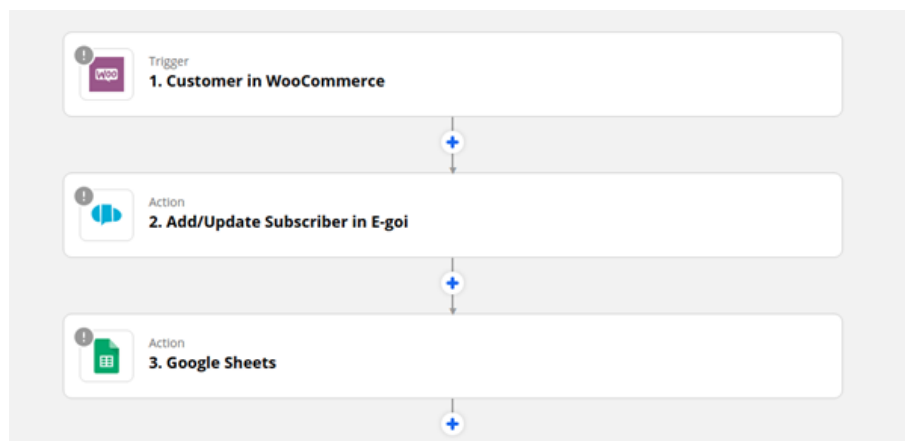


Figura 2.2: Exemplo de fluxo na plataforma Zapier

Planos

Existem quatro planos pagos dependendo das necessidades de cada cliente e um plano gratuito, como é visível na Figura 2.3.

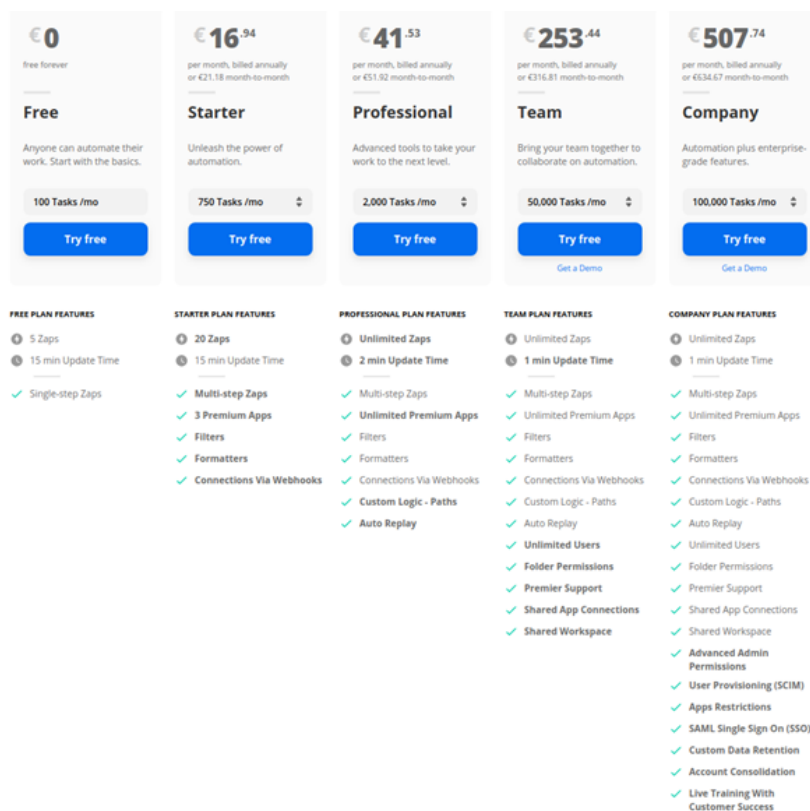


Figura 2.3: Planos da plataforma Zapier

O plano gratuito oferece apenas 100 execuções por mês e a criação de 5 *workflows*. As funcionalidades de etapas múltiplas e lógica condicional estão disponíveis a partir do primeiro plano pago, que apresenta 750 execuções mensais e a criação de 20 *workflows*. Os restantes planos pagos diferem principalmente no número de execuções mensais e em funcionalidades mais direccionadas para o suporte.

2.4.2 Microsoft Power Automate

O Power Automate [12] é um serviço *cloud* de criação de fluxos de trabalho automatizado entre diversas aplicações e serviços, criado pela Microsoft, em outubro de 2016. A criação destes fluxos permite simplificar processos de negócio e geri-los de uma forma mais eficiente.

Apesar do seu foco ser em processos de negócio, apresentam diversos conectores (que são partes que constituem o fluxo, podendo ser aplicações ou serviços) de fontes externas à Microsoft. Apresenta mais de 300 conectores para a criação de fluxos de automação, sendo que existe a facilidade de criar conectores personalizados, de forma a aumentar o leque disponível.

É uma ferramenta simples de utilizar e apresenta uma interface bastante intuitiva. Como é cloud-based apresenta prevenção de perda de dados e variadas ações para gerir dados como a filtragem, formatação, temporizador, condição, AI Builder, entre outras.

Ainda não existe uma integração E-goí disponível, no entanto, está na fase final do seu desenvolvimento, estando disponível brevemente. O fluxo de exemplo, Figura 2.4, é iniciado

quando é adicionado um evento no Google Calendar. O administrador é notificado por telemóvel e é enviado um e-mail pelo Outlook sobre esse evento.

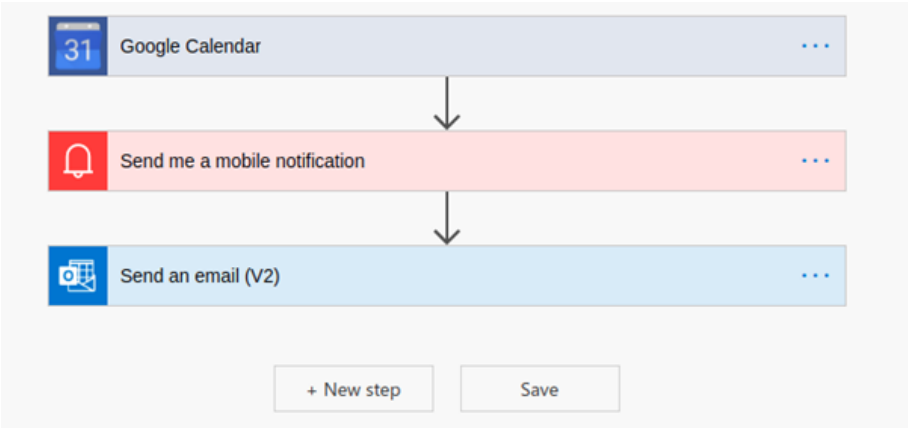


Figura 2.4: Exemplo de fluxo na plataforma Microsoft Power Automate

Planos

Existe um plano gratuito que apenas permite correr 750 fluxos por mês e apresenta limitações em certas funcionalidades e na disponibilidade de certos conectores. Na Figura 2.5, estão apresentados os 3 planos pagos que este serviço oferece.

	Per user plan €12.60 user/month	Per user plan with attended RPA €13.70 user/month	Per flow plan Starting at €421.50 per month
Run flows			
Active flows per user	Unlimited ²	Unlimited ²	N/A
Active attended RPA bots ³	--	1 bot ³	--
Active unattended RPA bots ³	--	€ per bot	€ per bot
AI Builder service credits ⁴	€	5,000/month	€
What's included			
Connect to your data	Use pre-built connectors		
	Use custom and on-premises connectors		
Store and manage data	Create and access custom entities		
Use Microsoft Dataverse (formerly Common Data Service)	50 MB database capacity ⁵		
	200 MB file capacity ⁵		

Figura 2.5: Planos da plataforma Microsoft Power Automate

Nos planos pagos não existe qualquer limitação relativamente ao número de execuções de fluxos. É permitido o uso de conectores *premium* (conectores apenas disponíveis em planos pagos). A grande diferença entre estes planos é a possibilidade de utilizar AI na criação de fluxos.

2.4.3 Integromat

O Integromat [13] é uma empresa de *software* que oferece uma ferramenta que permite conectar e automatizar fluxos baseados em triggers e actions. Cediada em Praga, República Checa, foi fundada por Tomáš Schel, Roman Bršlica, Ondřej Gazda, Pavel Duras, Patrik Šimek e Michal Toman, em 2016.

Apresenta mais de 500 aplicações e serviços prontos para serem utilizados na criação de fluxos simples ou complexos. Os fluxos podem apresentar infinitas etapas e a cada etapa podem ser utilizados filtros, funções e ciclos. Esta ferramenta diferencia-se por apresentar soluções adequadas para desenvolver fluxos complexos, pois apresenta módulos como *iterators*, *aggregators* e *routers*.

Os *iterators* são módulos que permitem dividir um conjunto de etapas num conjunto mais pequeno, permitindo que o fluxo não tenha de correr desde o início, se apenas é necessário que uma parte do fluxo seja executada novamente.

Os *aggregators* módulos que diferem dos *iterators* pois permitem unificar várias execuções numa só. Um exemplo da utilidade desta funcionalidade é a verificação de ficheiros existentes numa plataforma e o envio dos mesmos para um dado email. Com os módulos do tipo *aggregator*, quando são encontrados múltiplos ficheiros, ao invés de os enviar separadamente, os ficheiros são agregados e enviados numa só execução.

O módulo *router* permite duplicar dados, dividindo o fluxo em múltiplas rotas onde os dados serão processados de forma diferenciada.

Apesar de apresentar uma interface apelativa e simples, com um sistema de *drag-and-drop*, o desenvolvimento de certos fluxos pode tornar-se bastante complexo. Na Figura 2.6, está representado um possível fluxo desenvolvido no Integromat.



Figura 2.6: Exemplo de fluxo na plataforma Integromat. Fonte: [14]

Neste fluxo de exemplo, sempre que se recebe um novo email, será enviada uma mensagem para o Slack e qualquer ficheiro que esteja presente no email será guardado automaticamente na dropbox e o texto do e-mail guardado no Evernote.

Planos

O Integromat tem um plano gratuito que apresenta um limite de 1000 execuções de fluxos e um tamanho máximo de 100MB de transferência de dados por mês. Existem 4 planos pagos com limites superiores, como está representado na Figura 2.7.

FREE \$0 / month	BASIC \$9 / month	STANDARD \$29 / month	BUSINESS \$99 / month	PLATINUM \$299 / month	CUSTOM Need more?
1,000 Operations	10,000 Operations	40,000 Operations	150,000 Operations	800,000 Operations	800,000+ Operations
100 MB Data transfer	1 GB Data transfer	20 GB Data transfer	70 GB Data transfer	220 GB Data transfer	220+ GB Data transfer
15 minutes Minimum interval	5 minutes Minimum interval	1 minute Minimum interval	1 minute Minimum interval	1 minute Minimum interval	1 minute Minimum interval
Sign up	Sign up	Sign up	Sign up	Sign up	Let's connect

Figura 2.7: Planos da plataforma Integromat

O primeiro plano pago apresenta um limite de execuções de fluxos por mês bastante mais elevado que o plano gratuito, permitindo 10000 operações e um tamanho máximo de 1GB de transferência de dados por mês. Relativamente a cada plano pago difere o número de execuções mensais e o tamanho máximo de transferência de dados.

2.4.4 IFTTT

O IFTTT [15], conhecido por *if this then that* é um serviço que permite automatizar fluxos com declarações condicionais que se iniciam com *triggers*. Este serviço foi criado por Jesse Tane e Linden Tibbets, em 2011.

No IFTTT, o fluxo constituído por um *trigger* e uma ou mais ações é designado de *applet*. O uso deste serviço apresenta diferentes finalidades, como o uso pessoal para automatizar pequenas tarefas do dia a dia, de negócio com o produto IFTTT Business e para automatizar tarefas de casa com o produto IFTTT Smart Home que está preparado com diversas applets para automatizar tarefas, como por exemplo, ligar a luz quando é detectado movimento.

Apesar de apresentar mais de 600 aplicações e serviços, o IFTTT apresenta limitações quanto à complexidade dos seus fluxos. A criação de novos *applets* é possível mas apresenta poucas ferramentas, tornado a criação muito limitadora.

Planos

O IFTTT apresenta apenas um plano gratuito que permite criar três applets e outro plano pago sem qualquer limitação.

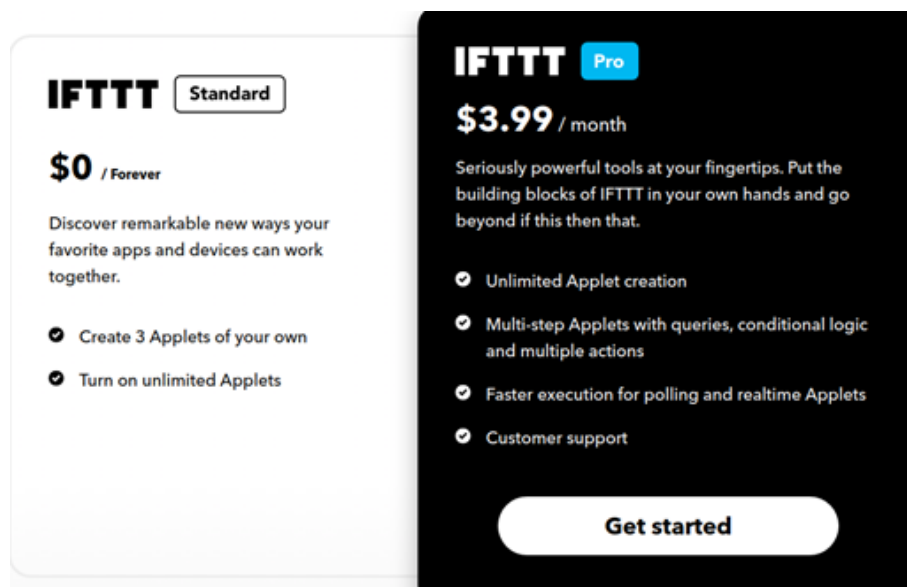


Figura 2.8: Planos da plataforma IFTTT

2.4.5 Automate.io

O Automate.io [16] é uma solução para a integração de aplicações e serviços que permite a criação de fluxos simples e complexos para automatizar vendas, marketing e processos de negócio.

Esta plataforma permite aos utilizadores integrar aplicações oferecendo uma interface *drag-and-drop*. Permite criar fluxos com múltiplas ações e é possível usar filtros e um temporizador a cada etapa do fluxo. Fluxos com ações e *triggers* são designados por *bot*.

Apresenta mais de 200 aplicações e serviços disponíveis para criar fluxos de integração e o seu ponto forte é a sua interface simples e intuitiva. A Figura 2.9 representa um fluxo simples com duas ações. Este fluxo, quando ativo, irá pesquisar por uma dada palavra ou frase e guardar todos os tweets que contenham essa palavra ou frase numa Google Sheets.

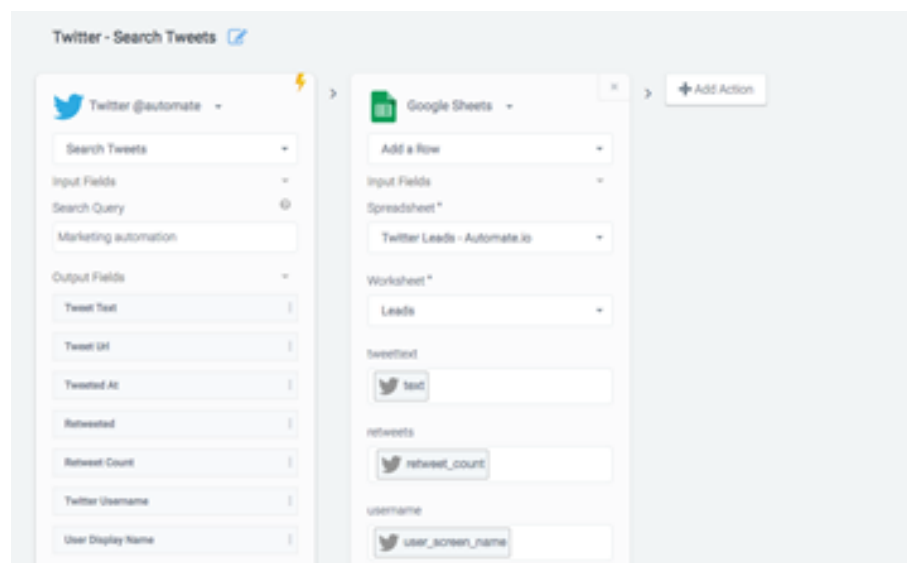


Figura 2.9: Exemplo de fluxo na plataforma Automate.io. Fonte: [17]

Planos

Relativamente aos planos, Figura 2.10, o Automate.io apresenta um plano gratuito com um limite de 300 execuções dos fluxos por mês e um limite de criação de 5 bots.

Free	Personal	Startup	Growth	Business	Enterprise
\$0/mo	\$29/mo	\$49/mo	\$99/mo	\$199/mo	\$399 onwards
Try for Free	Try Personal	Try Startup	Try Growth	Try Business	Try Enterprise
Actions: 300 Bots: 5 Data Check: 5 min	Actions: 2,000 Bots: 20 Data Check: 5 min	Actions: 10,000 Bots: 50 Data Check: 2 min	Actions: 30,000 Bots: 100 Data Check: 2 min	Actions: 100,000 Bots: 200 Data Check: 1 min	Actions: 500,000+ Bots: Unlimited Data Check: 1 min
✓ Single Action Bots	✓ Multi Action Bots ✓ Premium Apps	✓ Multi Action Bots ✓ Premium Apps ✓ Auto Retry	✓ Multi Action Bots ✓ Premium Apps ✓ Auto Retry ✓ Excess Actions	✓ Multi Action Bots ✓ Premium Apps ✓ Auto Retry ✓ Excess Actions ✓ Data Controls ✓ Team Account	✓ Multi Action Bots ✓ Premium Apps ✓ Auto Retry ✓ Excess Actions ✓ Data Controls ✓ Team Account

Figura 2.10: Planos da plataforma Automate.io

O primeiro plano pago apresenta um limite de 2000 execuções mensais, um limite de criação de 20 *bots* e acesso a aplicações e serviços *premium*, que são apenas acessíveis nos planos pagos. A principal diferença relativamente aos planos pagos é o limite de execuções mensais e as funcionalidades de gestão e suporte.

2.4.6 n8n.cloud

O n8n.cloud [18] é uma ferramenta de automação de *workflows* que permite automatizar processos de negócio e sincronizar dados entre aplicações e serviços. Esta plataforma é

baseada em nodes e é um SaaS (*software as a service*) que oferece a utilização do n8n.io [19].

O n8n.io é um projeto *fair-code* criado por Jan Oberhauser em 2019, que oferece a possibilidade de criar *workflows* de processos de negócio simples e complexos baseados em ações e *triggers*. Apresenta uma interface intuitiva e com um sistema de *drag-and-drop*. Este projeto é popular devido à sua fácil customização e criação de novos conectores.

O n8n apresenta cerca de 200 aplicações e serviços disponíveis para a criação dos fluxos. Existem módulos condicionais que oferecem a capacidade para criar fluxos mais complexos.

A Figura 2.10 representa um exemplo de um fluxo que envia uma receita todos os dias para o Telegram.

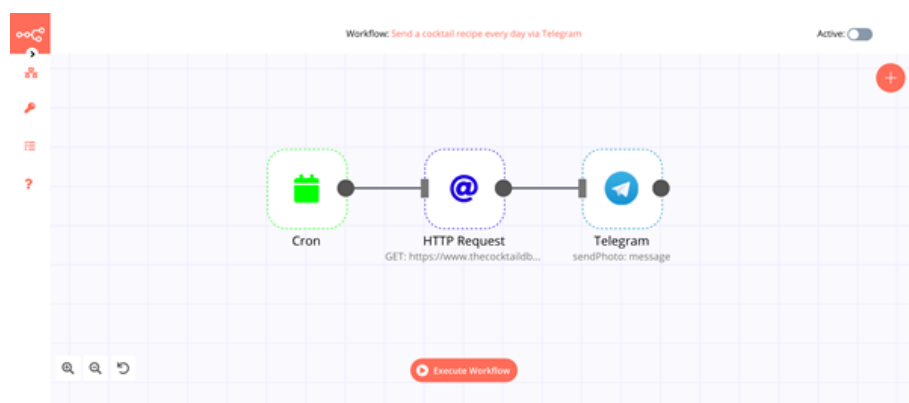


Figura 2.11: Exemplo de fluxo na plataforma n8n.cloud

Planos

Apresenta 3 planos pagos com diferentes limites no número de execuções por mês. O limite de execuções de fluxos do primeiro plano pago é de 5000, como é visível na Figura 2.12. Os restantes planos pagos apresentam um limite de 15000 e 60000 execuções de fluxos mensais.

	Start €20/ mo	Pro €50/ mo	Power €120/ mo
Workflow executions	5,000/ mo	15,000/ mo	60,000/ mo
Active workflows	20	100	200
Community support	✓	✓	✓
Priority support	-	-	✓

Figura 2.12: Planos da plataforma n8n.cloud

2.4.7 Workato

O Workato [20] é uma plataforma de integração e automação de *workflows* de processos de negócio. Foi fundada por Gautham Viswanathan, Harish Shetty, Dimitris Kogias e Vijay Tella, em 2013.

O Workato é reconhecido por disponibilizar mais de 500 aplicações e serviços e apresentar simplicidade e rapidez na criação e implementação de novos fluxos.

Oferece diferentes soluções para diferentes áreas como marketing, vendas, IT, finanças e recursos humanos. A presença de diferentes propriedades condicionais e a possibilidade de criar diversos caminhos para o fluxo proporciona a possibilidade de criar com facilidade fluxos complexos.

A Figura 2.13 representa um exemplo de um fluxo com ações condicionais. O fluxo em questão representa uma procura na Zendesk de uma organização, sempre que uma nova conta for adicionada numa conta Salesforce.



Figura 2.13: Exemplo de fluxo na plataforma Workato. Fonte: [21]

É de realçar que esta plataforma é ideal para grandes negócios e equipas grandes pois apresenta diversas funcionalidades como lógica condicional, *webhooks*, funcionalidades de gestão, segurança e escalabilidade.

Planos

Relativamente a planos, é necessário entrar em contacto para ser oferecido um plano à medida das necessidades de cada cliente.

2.4.8 Análise Comparativa

Após uma análise individual a cada uma das soluções apresentadas, é realizada uma análise comparativa, de forma a perceber as diferenças fundamentais entre elas.

A análise do conjunto de soluções apresentadas permitiu encontrar cinco atributos importantes para a sua diferenciação. Estes atributos são:

- Número de aplicações e serviços disponibilizados;
- Existência de *webhooks*;
- Possibilidade de usar uma lógica condicional na criação dos fluxos;
- Possibilidade de conectar serviços e aplicações com etapas múltiplas;
- Número de execuções por mês de fluxos, para uma versão gratuita.

Relativamente ao número de aplicações e serviços, é importante perceber se existe uma grande variedade de serviços disponíveis para criar fluxos de automação. No que diz respeito ao processo de criação dos fluxos, é identificado como essencial e motivo de análise, a existência de *webhooks* e a possibilidade de usar lógica condicional na criação dos fluxos, como a filtragem e formatação de dados. Como a criação de fluxos complexos é vista como um atributo-chave, analisa-se a possibilidade de conectar serviços e aplicações com etapas múltiplas. O último atributo refere-se à limitação de execuções de fluxos por mês, para um plano gratuito.

Pelo estudo individual de cada uma das soluções, foi possível perceber que existe um leque de funcionalidades que está presente na grande maioria das soluções. A criação de ações, criação de *triggers*, utilização de *webhooks*, filtragem ou formatação de dados são algumas das funcionalidades que são consideradas funcionalidades base para plataformas de integração e automação de fluxos.

Em contrapartida, existem funcionalidades que são exclusivas ou apenas estão presentes em algumas plataformas. Estas funcionalidades contemplam o acompanhamento de erros, interface *drag-and-drop*, utilização da AI que é exclusiva da plataforma Power Automate, entre outras.

A grande diferenciação entre plataformas de integração e automação de fluxos que apresentam as funcionalidades consideradas base, é feita considerando o limite de execuções mensais de fluxos.

A Tabela 2.1, representa a comparação entre as soluções analisadas utilizando os 5 parâmetros supramencionados.

Tabela 2.1: Comparação entre as diversas plataformas

	Aplicações	Webhooks	Lógica Condicional	Etapas Múltiplas	Execuções/mês
Zapier	200+	Sim	Sim	Com plano pago	100
n8n.cloud	200+	Sim	Sim	Sim	Não apresenta plano gratuito
Powe Automate	300+	Sim	Sim	Sim	750
Integromat	500+	Sim	Sim	Sim	1000
IFTTT	600+	Limitado	Limitado	Limitado	Ilimitado
Automate.io	200+	Sim	Sim	Com plano pago	300
Workato	500+	Sim	Sim	Sim	Não apresenta plano gratuito

Através da visualização e análise da Tabela 2.1, é possível verificar que existe um número de aplicações e serviços disponíveis para a criação de fluxos muito similar e compreendidos entre 200 a 600. De salientar, o elevado número de aplicações e serviços do Integromat, IFTTT e Workato. Relativamente a funcionalidades referentes à criação de ações e fluxos, na generalidade, todas apresentam possibilidade de utilização de *webhooks* e uso de lógica condicional. No entanto, em ambos os atributos, o IFTTT apresenta um uso limitado.

A criação de fluxos complexos depende bastante da possibilidade de conectar aplicações e serviços com etapas múltiplas. Na Tabela 2.1 é visível que plataformas como Zapier e Automate.io, não oferecem essa possibilidade num plano gratuito, sendo considerada uma limitação.

Relativamente ao limite de execuções mensais de fluxos para o plano gratuito, realça-se a inexistência de limitações para a plataforma IFTTT. Em contra partida, o Zapier e Automate.io apresentam um limite relativamente baixo, de 100 e 300, respetivamente. O Power automate apresenta um limite de 750 execuções mensais e o Integromat um limite de 1000 execuções mensais.

Realça-se a plataforma n8n.cloud pela utilização do projeto *fair-code* n8n.io que apresenta uma comunidade ativa e com uma constante evolução no número de conectores disponível. Considera-se que o projeto n8n.io é um projeto de bastante relevância e apresenta funcionalidades que podem colmatar bastantes necessidades da proposta desta dissertação.

De um ponto de vista mais alargado, considera-se que as plataformas apresentadas estão na sua maioria dentro de um mercado focado para a integração e automação de fluxos simples, complexos e direcionados para pequenas equipas ou uso pessoal. Apenas duas destas plataformas diferem deste mercado, a plataforma Workato e IFTTT. O Workato é

mais direcionado para grandes equipas e negócios, focado num processo empresarial onde é necessário criar fluxos extremamente complexos. Por outro lado, o IFTT apresenta-se como uma solução focada no uso pessoal e quotidiano, não oferecendo muitas funcionalidades para a criação de fluxos complexos.

Capítulo 3

Análise

Neste capítulo será desenvolvida a análise do projeto. Primeiramente são definidos os requisitos da nova solução. É também realizada um análise de valor.

3.1 Engenharia de Requisitos

A primeira medida de sucesso de um sistema de software é o grau de cumprimento do propósito para o qual o sistema foi criado. A engenharia de requisitos é o processo de descoberta desse propósito, identificando as partes interessadas e as suas necessidades de uma forma que seja passível de analisar, comunicar e implementar [22].

Nesta secção serão apresentados os requisitos funcionais e não funcionais.

3.1.1 Requisitos Funcionais

Os requisitos funcionais são todos os problemas e necessidades que devem ser atendidos e resolvidos pelo software. Primeiramente, identificam-se todos os casos de uso e representam-se através de um diagrama de casos de uso. Na Figura 3.1, estão representados os casos de uso identificados.

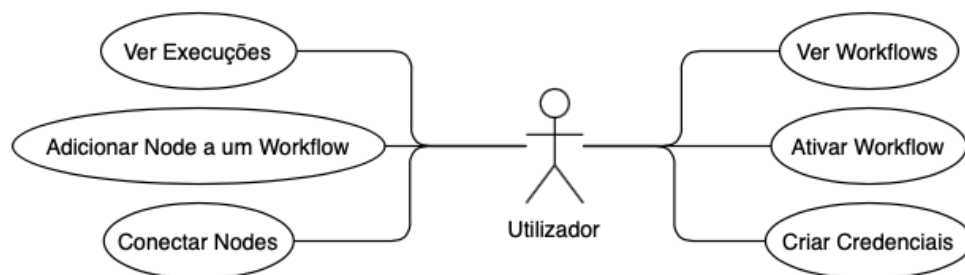


Figura 3.1: Diagrama de Casos de Uso

Foi identificado um ator, o utilizador Goidini, e os casos de uso que lhe estão associados. Estes casos de uso são descritos no decorrer nesta secção.

UC01 - Adicionar um Node ao Workflow

Ator: Utilizador

Descrição

Para todos os clientes que tenham acesso à plataforma, deverá ser possível adicionar um *node* a um *workflow* de integrações. O resultado dessa criação, é um novo *node* presente na área de criação do *workflow*.

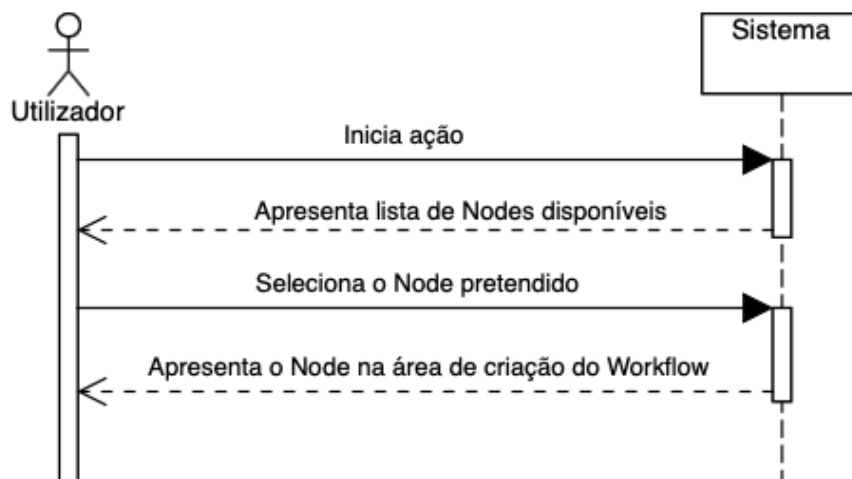


Figura 3.2: SSD do UC01

Considera-se o seguinte fluxo de eventos:

1. O utilizador inicia a adição de um novo *node*.
2. O sistema obtém e apresenta a lista de *nodes* disponíveis. Os *nodes* podem ser filtrados por *regular node* e *trigger node*. A principal diferença é a sua utilização no *workflow*. O *regular node* é utilizado em qualquer fase do *workflow* enquanto que o *trigger node* é utilizado apenas no início do *workflow*.
3. O utilizador seleciona o *node* pretendido.
4. O sistema apresenta o *node* selecionado na área de criação do *workflow*.

UC02 - Criar Credenciais

Ator: Utilizador

Descrição

Para todos os clientes que tenham acesso à plataforma, deverá ser possível criar credenciais de acesso a cada *node*. O resultado dessa criação, é uma credencial que dá acesso ao *webhook* associado ao *node*.

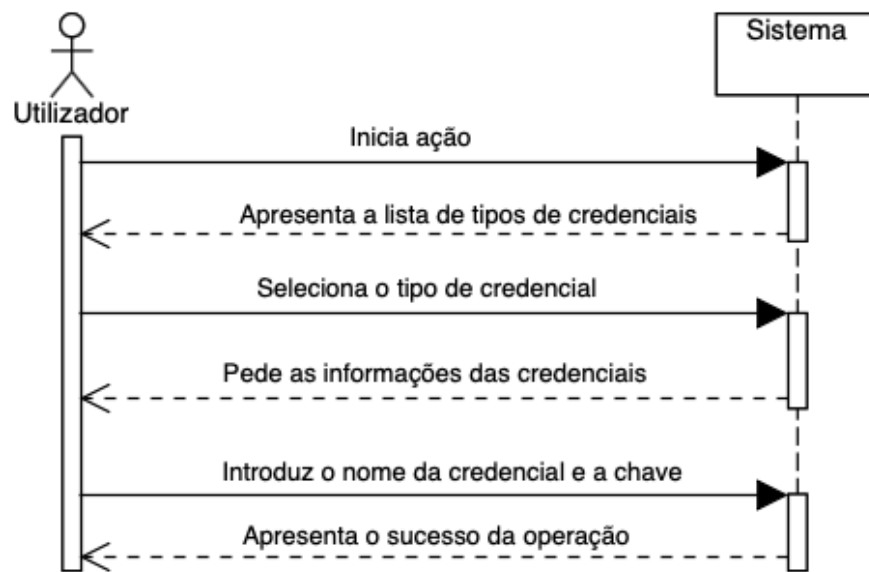


Figura 3.3: SSD do UC02

Considera-se o seguinte fluxo de eventos:

1. O utilizador inicia de criação de credenciais.
2. O sistema apresenta a lista de tipos de credenciais. O tipo de credencial é referente ao *node* ao qual a credencial dá acesso de utilização.
3. O utilizador seleciona o tipo de credencial.
4. O sistema pede as informações necessárias para a criação de credenciais. Estas informações são o nome e a chave de acesso.
5. O utilizador introduz o nome e a chave da credencial.
6. O sistema valida e guarda a credencial apresentando o sucesso da criação.

UC03 - Configurar Node E-goi

Ator: Utilizador

Descrição

Para todos os cliente que tenham acesso à plataforma, deverá ser possível selecionar e configurar o *node* E-goi. O resultado dessa ação é o *node* E-goi configurado na área de criação de workflows.

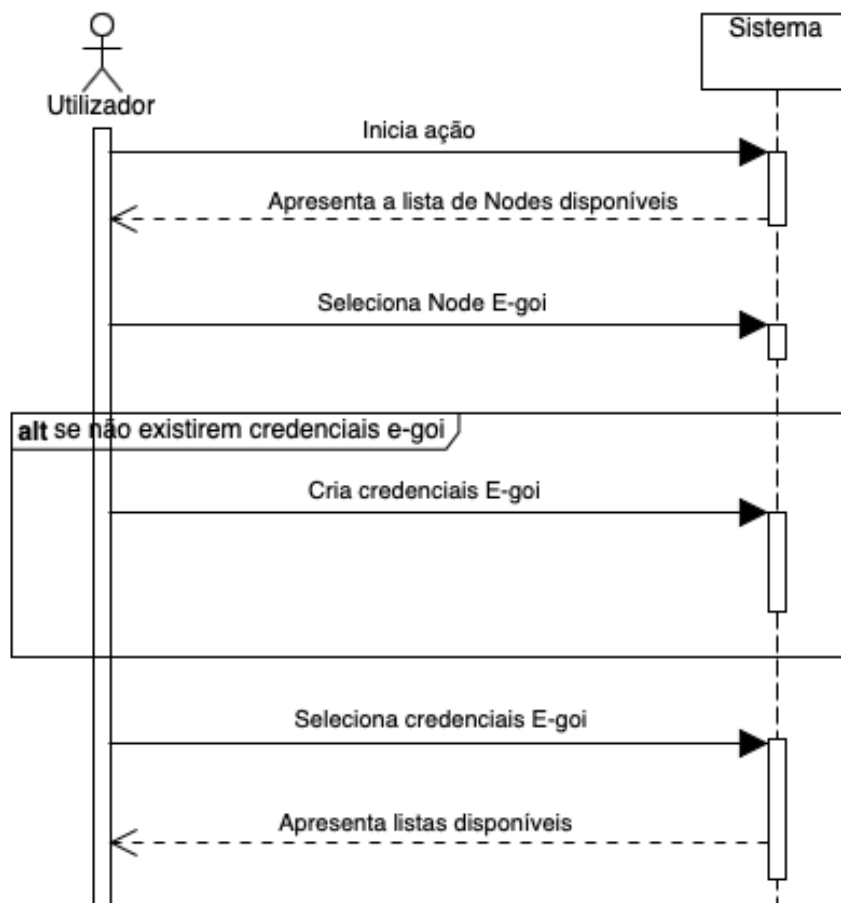


Figura 3.4: SSD do UC03

Considera-se o seguinte fluxo de eventos:

1. O utilizador inicia a ação para visualizar a lista de *nodes* disponíveis.
2. O sistema obtém e apresenta a lista de *nodes*.
3. O utilizador seleciona o *node* E-goi.
4. Caso não exista nenhuma credencial do tipo E-goi então o utilizador necessita de criar. No caso de ser uma credencial válida o sistema guarda a credencial.
5. O utilizador seleciona a credencial E-goi.
6. O sistema obtém os dados necessários para conectar com outros *nodes* e apresenta ao utilizador.

UC04 - Conectar Nodes

Ator: Utilizador

Descrição

Para todos os clientes que tenham acesso à plataforma, deverá ser possível conectar dois *nodes* que estejam presentes na área de criação do *workflow*. O resultado dessa criação, é um fluxo com passagem de informação entre dois *nodes*.

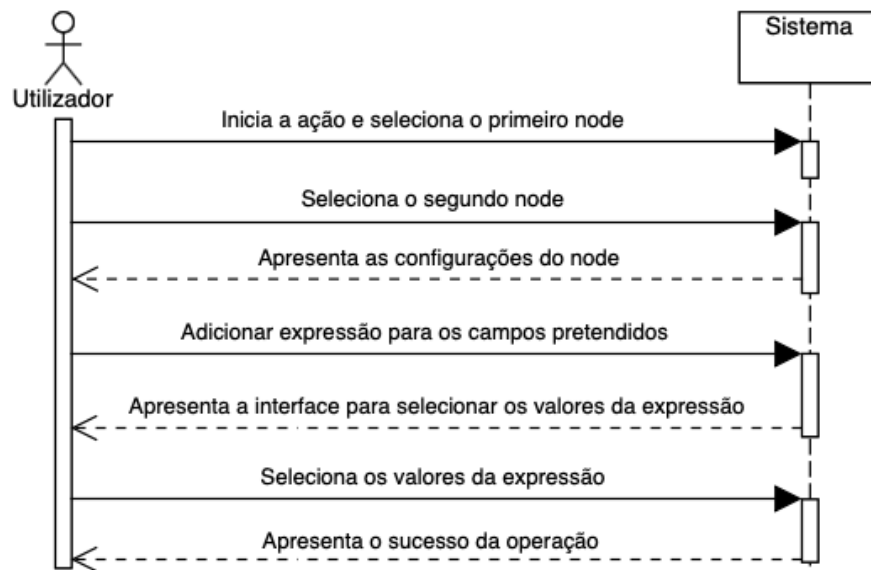


Figura 3.5: SSD do UC04

Considera-se o seguinte fluxo de eventos:

1. O utilizador seleciona o primeiro node a conectar.
2. O utilizador seleciona o segundo *node* de forma a abrir a área de configuração do *node*.
3. O sistema apresenta as configurações do *node*.
4. O utilizador seleciona os campos que quer preencher com informações vindas do *node* anterior. Adiciona uma expressão para esses campos.
5. O sistema apresenta a interface para selecionar os valores que quer utilizar para a criação da expressão. Apresenta também a lista de valores que o *node* anterior passa.
6. O utilizador seleciona os valores para cada campo.
7. O sistema apresenta o sucesso da operação e guarda a expressão.

UC05 - Ativar Workflow

Ator: Utilizador

Descrição

Para todos os clientes que tenham acesso à plataforma, deverá ser possível ativar um *work-flow* de integrações. O resultado dessa criação, é um fluxo ativo e funcional.

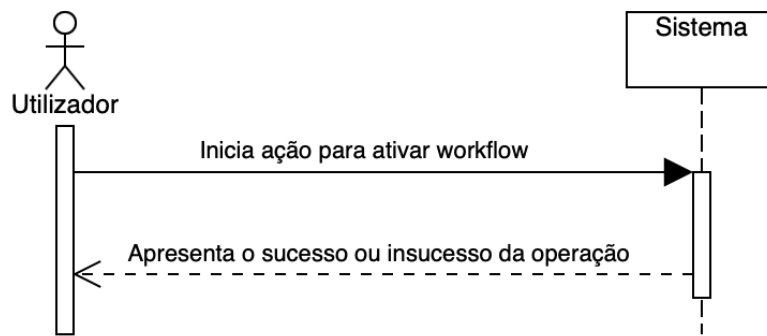


Figura 3.6: SSD do UC05

Considera-se o seguinte fluxo de eventos:

1. O utilizador inicia ação para ativar um *workflow*.
2. O sistema verifica se é possível ativar o *workflow* e apresenta o sucesso da ativação.

UC06 - Ver Execuções

Ator: Utilizador

Descrição

Para todos os clientes que tenham acesso à plataforma, deverá ser possível ver todas as execuções de um determinado *workflow*. O resultado dessa escolha é uma lista de execuções do *workflow* escolhido pelo utilizador.

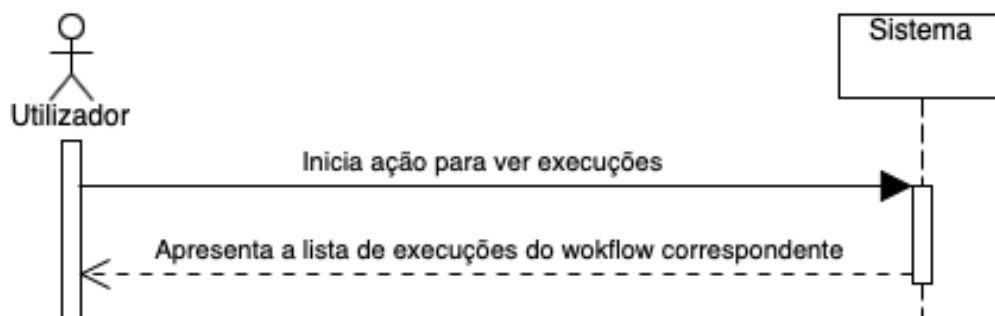


Figura 3.7: SSD do UC06

Considera-se o seguinte fluxo de eventos:

1. O utilizador inicia ação para ver as execuções do *workflow*.
2. O sistema apresenta a lista de execuções do *workflow*.

UC07 - Ver Workflows

Ator: Utilizador

Descrição

Para todos os clientes que tenham acesso à plataforma, deverá ser possível visualizar a lista de *workflows* existentes. O resultado dessa ação, é a apresentação da lista de *workflows*.

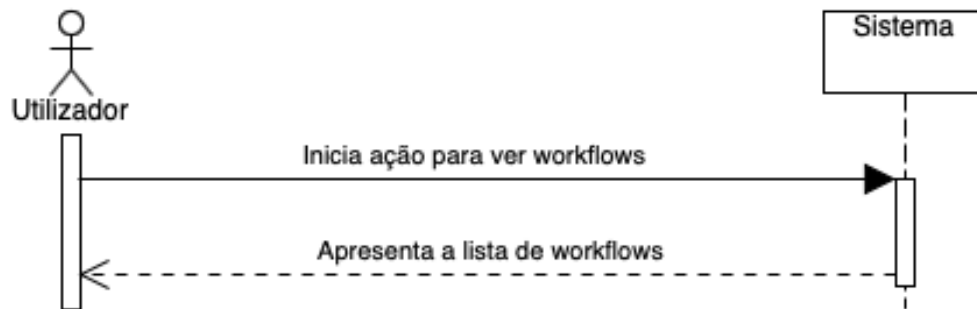


Figura 3.8: SSD do UC07

Considera-se o seguinte fluxo de eventos:

1. O utilizador inicia ação para ver os *workflows* existentes.
2. O sistema apresenta a lista de *workflows* com a data de criação dos mesmos e com informação do estado de ativação.

3.1.2 Requisitos Não Funcionais

Os requisitos não funcionais foram levantados com o auxílio do modelo FURPS+.

O FURPS+ é um acrónimo para Funcionalidade, Usabilidade, Confiabilidade, Desempenho e Suportabilidade. Este acrónimo representa um modelo de classificação de atributos de qualidade [23]. Os requisitos estão listados na Tabela 3.1.

Tabela 3.1: Tabela FURPS+

FURPS+	Descrição do Requisito
Funcionalidade	O registo e login na plataforma, deve ser realizado através do gestor de utilizadores do Goidini
Funcionalidade	O email de registo e login terá de ser de uma conta E-goi existente
Usabilidade	O sistema deve ser preventivo relativamente a erros de usabilidade do utilizador
Usabilidade	O utilizador deve adicionar um novo node com um máximo de 3 cliques
Confiabilidade	O sistema deve fazer gravações automáticas periódicas
Desempenho	O tempo de criação de um workflow deve ser aceitável para o utilizador
Suportabilidade	O sistema deve ser desenvolvido de forma a ser facilmente escalável
+	O sistema deve ser desenvolvido usando a framework Vue.js, a linguagem de programação Typescript e o sistema de base de dados MariaDB
+	O sistema deve respeitar a licença de fair-code do projeto n8n

3.2 Análise de Valor

No presente sub-capítulo é feita uma análise de valor do projeto descrito neste documento. No decorrer deste sub-capítulo são abrangidos os conceitos de valor e inovação.

A SAVE (Society of American Value Engineers)[24], define a análise de valor como uma aplicação sistémica de técnicas reconhecidas que identificam as funções de um produto ou serviço, estabelecem um valor para essas funções e proporcionam as funções necessárias ao mais baixo custo.

O desenvolvimento de um novo produto exige um estudo sobre a aceitação por parte de possíveis clientes e sobre o risco económico associado. Esta necessidade de estudo, é contemplada com os processos de inovação que permitem garantir a uma organização que a implementação de um novo produto não é feita de forma a proporcionar desvantagens ou elevados custos e recursos.

3.2.1 Processo de Inovação

A inovação consiste na geração de uma nova ideia e da sua implementação gerando um novo produto, processo ou serviço que levará a um dinâmico crescimento económico da nação e à

criação de lucro para uma organização. Inovação é um longo processo de inúmeras tomadas de decisão organizacional [25].

O processo de inovação pode ser dividido em três partes [26]:

- Fuzzy Front End;
- New Product Development;
- Comercialização.

O front-end da inovação, maioritariamente denominado por Fuzzy Front End (FFE), apresenta uma das grandes oportunidades para melhorar, geralmente, o processo de inovação [27]. A atenção nas atividades de front-end precede o estruturado processo, de modo a aumentar o valor, a quantidade e a probabilidade de sucesso de conceitos, pela introdução do desenvolvimento do produto e da sua comercialização [28].

A comparação das práticas de FFE em diversas organizações, permitiu concluir que existe uma diferença nas definições e termos comuns relativos a elementos do FFE [29]. De forma a contemplar a inexistência de uma linguagem bem definida, levou Koen a criar o New Concept Development (NCD) [28].

O NCD consiste na implementação dos conceitos do FFE. Trata-se da criação de um produto com base no conceito e na análise do nicho de mercado, dos possíveis clientes e na sua satisfação [29].

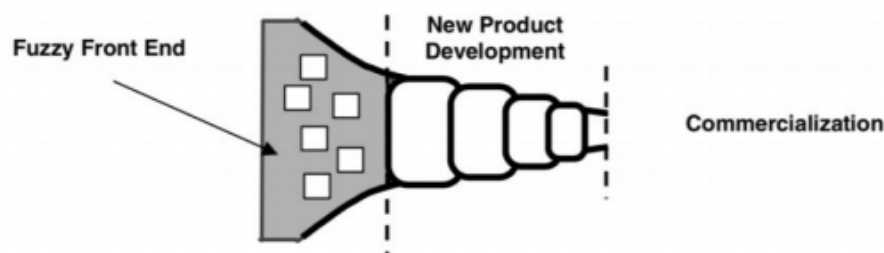


Figura 3.9: Modelo de processo de valor. Fonte: [30]

O último passo do processo de inovação refere-se à Comercialização, cujo objetivo é vender o produto criado no New Product Development (NPD). Nesta etapa o produto já foi testado e prossegue para a fase de produção e comercialização.

3.2.2 New Concept Development

De forma a proporcionar melhor clareza ao processo de inovação Koen desenvolveu o New Concept Development [27]. O modelo desenvolvido, Figura 3.10, é composto por três partes principais: o motor, os cinco elementos e os fatores de influência.

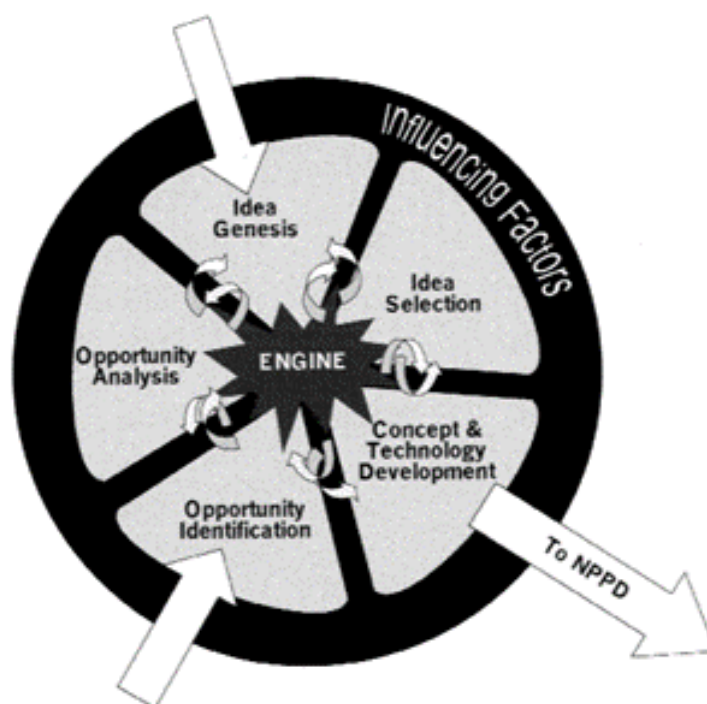


Figura 3.10: Modelo de New Concept Development. Fonte: [30]

O motor representa a liderança, cultura e estratégias de negócio de uma organização. Os cinco elementos representados são: identificação de oportunidade, análise da oportunidade, geração da ideia, seleção da ideia e definição do conceito. Por último, os fatores de influência consistem na capacidade organizacional e fatores internos e externos que possam estar envolvidos e que são incontrolláveis pela organização [30].

O modelo representado na Figura 3.10 tem uma forma circular para sugerir que é expectável que as ideias fluam, circulem e itelem entre todos os cinco elementos.

Identificação da Oportunidade

A identificação da oportunidade é a primeira fase do NCD e é geralmente motivada pelos objetivos da organização e do negócio. As novas oportunidades podem estar relacionadas com fatores internos ou externos.

Na E-goi, a identificação da oportunidade é motivada pelas necessidades dos utilizadores, priorizando o constante melhoramento dos produtos. Como tal, os problemas identificados na Secção 1.2 do Capítulo 1, representam a oportunidade existente para melhorar a automação de integrações, de forma a existir uma plataforma no E-goi que permita criar *workflows* com diversas aplicações e serviços.

Após ter sido identificada a oportunidade, é necessário analisar o seu valor para confirmar que é vantajoso perseguir a oportunidade.

Análise da Oportunidade

De forma a tirar conclusões relativamente ao valor que a oportunidade pode acrescentar ao produto, são necessárias informações adicionais, que envolvem um planeamento realizado

com antecedência. São realizadas através de estudos de mercado e experiências científicas. A análise da oportunidade deve ser parte integrante de um processo formal e incremental [30].

Uma análise completa deve incluir:

Enquadramento estratégico: Determinar se a oportunidade se enquadra nas forças, fraquezas e ameaças do mercado onde a organização se insere.

Avaliação do segmento de mercado: Descrição do mercado e análise de fatores de mercado como fatores económicos, tecnológicos e culturais.

Análise de concorrência: Determinar os maiores concorrentes presentes no segmento de mercado identificado.

Avaliação do cliente: Determinar quais são as necessidades dos clientes que pretendem ser satisfeitas pela criação do novo produto.

Enquadramento Estratégico

Relativamente ao enquadramento estratégico, de forma a determinar com clareza, as forças, fraquezas e ameaças é realizada uma análise SWOT. O gráfico de análise SWOT estão permite situar a oportunidade no mercado, identificando os fatores internos e externos associados.



Figura 3.11: Análise SWOT

A análise SWOT apresentada, na Figura 3.11, representa a análise feita à oportunidade de criação de um sistema de automação de workflows e integrações. Foram identificados três pontos relativos às forças. A criação da plataforma de automação de workflows e integrações

integrada no E-goi proporciona vantagens relativamente à atual solução e aos concorrentes. O facto de ser criado um ponto único que possibilita ao cliente criar fluxos de forma simples. A criação desta solução também permite aumentar o número de integrações existente no E-goi, sendo um ponto fulcral para o sucesso do projeto.

O aumento na utilização de integrações e do produto E-goi é uma oportunidade importante e fundamental que foi identificada no decorrer da análise SWOT. A esta oportunidade acresce a possibilidade de maior retenção de clientes devido à existência de uma plataforma de automação de workflows.

Relativamente às fraquezas, foram identificadas duas grandes fraquezas. A primeira traduz-se na dificuldade adjacente ao desenvolvimento de uma nova funcionalidade num sistema existente, sendo necessário identificar o risco associado. O desenvolvimento da solução apresenta também um custo elevado, pois é necessário desenvolver, implementar e testar a solução recorrendo a recursos tecnológicos da organização.

Por último, as ameaças identificadas estão interligadas uma vez que se referem à atuação dos concorrentes no mercado. O mercado de automação de *workflows* está repleto de soluções com finalidades distintas e as principais plataformas apresentam um leque vasto de integrações disponíveis, sendo uma ameaça ativa no mercado.

Avaliação do segmento de mercado

O segmento de mercado considerado é o mercado de automação de *workflows* e integração. Este mercado está em ascensão nos últimos cinco anos, como é visível na Figura 3.12, que representa o gráfico de tendências do Google Trends e permite verificar esta necessidade crescente de automatizar processos tanto para finalidades pessoais ou empresariais.

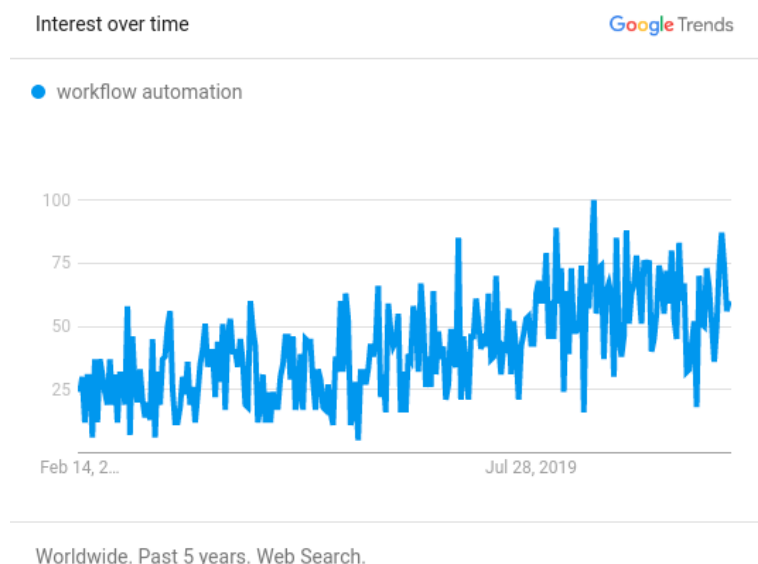


Figura 3.12: Gráfico de pesquisas sobre "Workflow Automation" entre 2016 e 2021

De forma a apresentar uma avaliação do segmento de mercado é necessário estudar a atuação da concorrência no mercado considerado. Através da análise presente na Secção 2.4.8 do Capítulo 2 foram identificados dois concorrentes principais. A plataforma Zapier e Integromat.

O Zapier apresenta uma ascensão relativamente ao número de utilizadores e à receita produzida em cada ano. No ano de 2014 esta plataforma apresentava cerca de 400 mil utilizadores e uma receita anual de cerca de 5 milhões de dólares. Em 2018 conseguiu registar cerca de 3 milhões de utilizadores e uma receita anual de 50 milhões de dólares [31].

O Integromat, sendo uma plataforma mais recente, em 2018 apresentava 77 mil utilizadores, em 2019 esse número subiu para 205 mil [32] e em 2020 apresentou cerca 449 mil utilizadores ativos da plataforma [33].

Análise de concorrência

Os principais concorrentes foram identificados na avaliação do segmento de mercado, e são o Zapier e Integromat.

As principais vantagens que a oportunidade ao qual se realiza esta análise apresenta são o facto de estar integrado com o produto E-goi co-existindo no mesmo ambiente. Outra das vantagens é o facto de não apresentar limitações quanto às execuções mensais dos fluxos que existe nas diversas plataformas.

Avaliação do cliente

A oportunidade está direccionada para clientes que necessitem de uma plataforma que permita automatizar workflows de integrações e que utilizem o produto E-goi. Com o desenvolvimento da oportunidade suprimem-se uma das problemáticas identificadas na Secção 1.1 do Capítulo 1 que é a crescente demanda dos clientes por serviços mais *ready-to-go*.

Geração de Ideias

A geração de uma ideia consiste no seu nascimento, desenvolvimento e maturação. O processo é evolutivo e as ideias são levantadas, recusadas, combinadas, modificadas e evoluídas. [30]

Assim, após uma análise da oportunidade e de um *brainstorm* e discussão interna surgiram três abordagens distintas:

1. Implementação de uma plataforma no Goidini, reestruturando o projeto *fair-code* n8n.io e utilizando o sistema de gestão de utilizadores do Goidini.
2. Implementação de uma plataforma dentro dos BO's / Conta E-goi, reestruturando o projeto *fair-code* n8n.io.
3. Implementação de uma plataforma no Goidini, totalmente independente (plataforma, lógica e conectores).

A utilização do n8n.io foi uma alternativa identificada por apresentar diversas vantagens. A primeira, por ser um projeto *fair-code*. E a segunda por já apresentar mais de 200 aplicações e serviços para serem utilizadas na criação de novos fluxos.

O problema que as diferentes abordagens pretendem resolver, é comum apesar de diferirem bastante na sua implementação. Como tal, torna-se imperativo realizar uma análise recorrendo a critérios chave que serão utilizados para auxiliar na escolha da melhor abordagem a seguir.

Seleção de Ideias

A seleção de ideias é considerada importante e complicada para as organizações, pois são necessárias diversas fases de análise, para perceber qual é a ideia que poderá trazer mais valor para a organização [30].

É necessário perceber quais são as vantagens e desvantagens de cada abordagem, de forma a ser realizada uma escolha certa. É importante ter em conta que o custo e esforço da implementação das diferentes ideias pode variar.

Como se trata de uma decisão difícil e que envolve diversos parâmetros, é utilizado um método de apoio à tomada de decisão. O método utilizado nesta dissertação é o Analytic Hierarchy Process (AHP). [34]

O método divide-se em cinco fases: Divisão Hierárquica, Estabelecer Prioridades Relativas, Avaliar Consistência das Prioridades Relativas, Construir Matriz da Comparação Paritária e Escolha da Alternativa.

Divisão Hierárquica

A divisão hierárquica consiste na decomposição do problema numa hierarquia composta por um objetivo, critérios e alternativas.

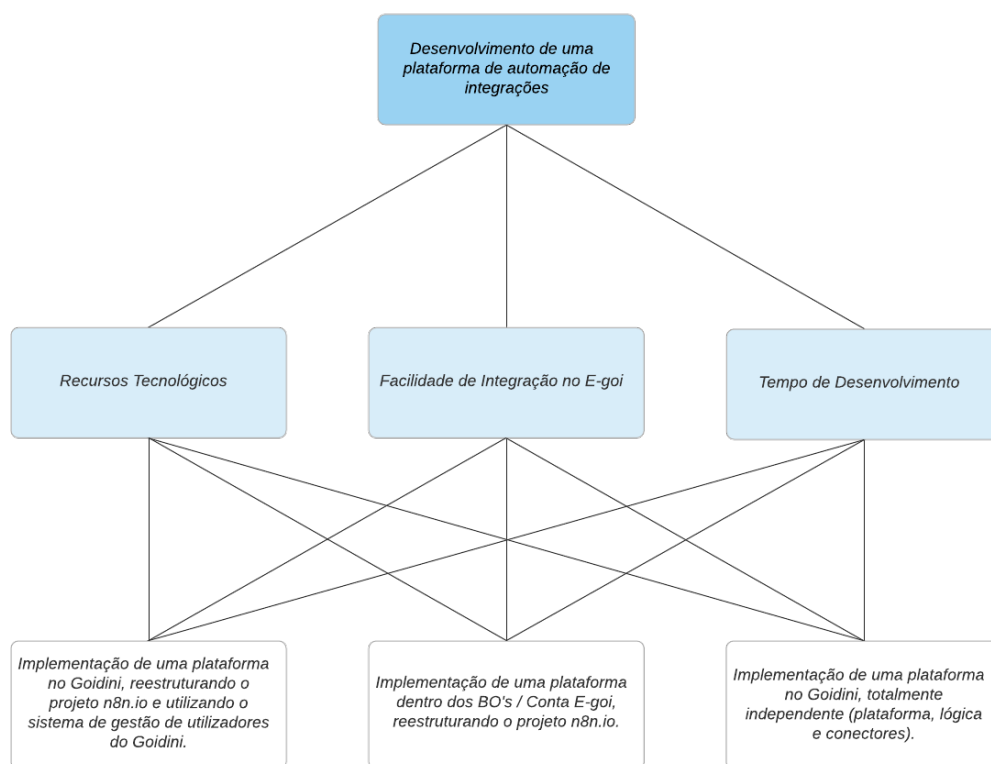


Figura 3.13: Estrutura Hierárquica do AHP

O objetivo definido é o desenvolvimento de uma plataforma de automação de integrações.

De forma a atingir este objetivo foram identificados os três critérios mais relevantes: os recursos tecnológicos (servidores, softwares e tecnologias) necessários para o desenvolvimento da ideias, a facilidade de integração no E-goi e o tempo de desenvolvimento.

As soluções são as abordagens identificadas na Secção 3.2.2 por atingirem o objetivo definido.

Estabelecer Prioridades Relativas

É necessário definir prioridades entre os diferentes critérios. Para se fazer a priorização dos critérios, é utilizada a escala fundamental [35] que está disponível no Anexo A.1.

Define-se:

A: Recursos Tecnológicos;

B: Facilidade de Integração;

C: Tempo de Desenvolvimento;

Tabela 3.2: Matriz de comparação dos critérios do segundo nível

	A	B	C	Prioridades Relativas
A	1	2	3	0.54
B	0.5	1	2	0.30
C	0.33	0.5	1	0.16

De forma a obter as prioridades relativas apresentadas na tabela de matriz de comparação dos critérios do segundo nível, é necessário normalizar a matriz e calcular a média aritmética dos valores de cada linha da matriz normalizada. A matriz normalizada está apresentada no Anexo A.3.

Analisando a Tabela A.5, é perceptível que os recursos tecnológicos são um pouco mais importantes que a facilidade de integração e são igualmente importantes como o tempo de desenvolvimento.

De acordo com os valores das prioridades relativas, é possível afirmar que os recursos tecnológicos são o critério mais importante, seguido da facilidade de integração e do tempo de desenvolvimento, respetivamente.

Após ter sido realizada a priorização dos critérios, é necessário verificar a sua consistência uma vez que a priorização é subjetiva.

Avaliar Consistência das Prioridades Relativas

Para avaliar a consistência das prioridades relativas, é necessário calcular a Razão de Consistência (RC) dos critérios.

No Anexo A.4, pode-se observar todos os cálculos realizados para obter a RC. Caso seja menor do que 0.1, os valores são considerados consistentes. O valor obtido:

$$RC \simeq 0.009 < 0.1$$

Assim, pode-se afirmar, que os valores das prioridades relativas se encontram consistentes.

Construir Matriz da Comparação Paritária

Após ter sido verificada a consistência dos valores de prioridades relativas, é feita a priorização das soluções relativamente a cada critério. De forma a simplificar, as matrizes normalizadas encontram-se no Anexo A.5.

Define-se:

X: Implementação de uma plataforma no Goidini, reestruturando o projeto fair-code n8n.io e utilizando o sistema de gestão de utilizadores do Goidini;

Y: Implementação de uma plataforma dentro dos BO's / Conta E-goi, reestruturando o projeto fair-code n8n.io;

Z: Implementação de uma plataforma no Goidini, totalmente independente (plataforma, lógica e conectores);

Tabela 3.3: Matriz de comparação paritária de recursos tecnológicos

	X	Y	Z	Prioridades Relativas
X	1	2	5	0.56
Y	0.5	1	4	0.33
Z	0.2	0.25	1	0.13

Tabela 3.4: Matriz de comparação paritária de facilidade de integração

	X	Y	Z	Prioridades Relativas
X	1	2	4	0.56
Y	0.5	1	3	0.32
Z	0.25	0.33	1	0.12

Tabela 3.5: Matriz de comparação paritária de tempo de desenvolvimento

	X	Y	Z	Prioridades Relativas
X	1	1	6	0.46
Y	1	1	6	0.46
Z	0.17	0.17	1	0.08

Através da análise das três tabelas conclui-se que para os critérios de recursos tecnológicos e facilidade de integração, a abordagem X é bastante mais convincente. Na matriz de comparação paritária do tempo de desenvolvimento, observa-se um igual valor de benefício para as abordagens X e Y. A abordagem Z é a opção menos benéfica relativamente a qualquer critério.

Escolha da Alternativa

Após a conclusão das etapas anteriores, verificam-se as seguintes matrizes:

Tabela 3.6: Matriz de prioridades relativas das abordagens

	A	B	C
X	0.56	0.56	0.46
Y	0.33	0.32	0.46
Z	0.13	0.12	0.08

Tabela 3.7: Matriz de prioridades relativas dos critérios

	Prioridade Relativa
X	0.54
Y	0.30
Z	0.16

A multiplicação da matriz representada na Tabela 3.6 com a matriz representada na Tabela 3.7 origina a prioridade composta das alternativas:

Tabela 3.8: Matriz de prioridade composta

	Prioridade Composta
X	0.54
Y	0.34
Z	0.12

Através da análise da Tabela 3.8, verifica-se que a abordagem que obteve um maior resultado foi a abordagem X. Conclui-se que a implementação de uma plataforma no Goidini, reestruturando o projeto *fair-code* n8n.io e utilizando o sistema de gestão de utilizadores do Goidini, é a melhor abordagem para a resolução do problema.

Definição do Conceito

O conceito para a automação de integrações consiste num sistema de *drag-and-drop* onde é possível criar *workflows* de integração arrastando *triggers* e ações para uma área destinada à criação dos fluxos. Deve existir uma área para listar as aplicações e serviços disponíveis. A interface deve ser simples e intuitiva.

3.2.3 Valor da Solução

Valor, Valor para o Cliente, Valor Percebido

O conceito de valor é usado de diversas formas. Em muitos casos, o valor para a organização é diferente do valor para o cliente [36]. No entanto, é importante definir, numa primeira instância, o valor de um produto. Este consiste na visão do cliente na relação entre os benefícios que o produto oferece e o custo do produto [37]. Deve existir uma preocupação constante dentro de uma organização para oferecer ao cliente um produto com valor.

Para o cliente, o valor está relacionado com a sua percepção ou experiência com um produto ou serviço, gerando uma conexão psicológica ou emocional. Uma vez que tem uma base subjetiva, o valor para o cliente é diferente para cada indivíduo [38]. Relativamente à dissertação, o facto de se oferecer uma plataforma com uma interface simples e intuitiva dentro do produto E-goi, pode gerar valor para o cliente por ter uma experiência agradável na sua utilização.

O valor percebido é considerado o indicador mais importante quando se refere a intenção de recompra de um produto por parte de um cliente [39]. Com o reconhecimento da importância do valor percebido, surge o reconhecimento por parte das organizações para criar e oferecer uma boa experiência para o cliente. No âmbito do projeto, a oferta de uma plataforma de automação de integrações sem limitações quanto às execuções de fluxos por mês, oferece ao cliente um enorme valor.

Proposta de Valor

Uma proposta de valor pode ser entendida como as declarações de benefícios que são entregues por uma organização aos seus constituintes externos [40]. É também entendida, como a forma como um produto ou serviço é preparado e oferecido para satisfazer as necessidades do consumidor [41].

Relativamente ao projeto, a proposta de valor consiste na oferta de uma plataforma de automação de *workflows* de integrações que se destaca pela possibilidade de criação de fluxos complexos com uma interface simples e intuitiva e que não apresenta limites mensais de execuções de fluxos, diferenciando-se dos concorrentes.

3.2.4 Business Model Canvas

O Business Model Canvas é um modelo composto por nove blocos interligados que permitem discutir e analisar novas propostas de negócio. A análise permite verificar como a organização gera valor para os clientes [42]. O modelo Canvas do projeto desta dissertação está descrito na Figura 3.14.

Parceiros Chave	Atividades Chave	Proposta de Valor	Relação com Clientes	Segmentos de Clientes
	Recursos Chave		Canais	
	Desenvolvimento e implementação da plataforma	- Criação de uma plataforma de automação de workflows de integrações; - Interface simples e intuitiva;	Suporte técnico;	- Pequenas e médias empresa; - Indivíduos interessados em automatizar o seu trabalho ou vida pessoal.
	- Projeto <i>fair-code</i> n8n; - Conhecimento tecnológico.		- Plataforma E-goi; - Blog E-goi.	
Custos			Fonte de Retorno	
- Hardware; - Equipas de desenvolvimento.			Uso da plataforma E-goi.	

Figura 3.14: Modelo Canvas

Os nove blocos representados são:

Parceiros Chave: Representam a rede de fornecedores e/ou parceiros essenciais para a manter o funcionamento do projeto. No âmbito do projeto, não existe nenhuma rede de fornecedores a considerar.

Atividades Chave: Este bloco representa as atividades necessárias para criar a proposta de valor. No âmbito do projeto, as atividades contemplam o desenvolvimento de uma arquitetura para a automação de workflows e a sua implementação;

Relação com Clientes: Representa o tipo de relação que é estabelecida com cada segmento de clientes. No âmbito do projeto, a E-goi fornece todos os meios necessários para que o cliente seja capaz de se auto-ajudar, bem como suporte técnico sempre que necessário;

Segmentos de Clientes: Este bloco representa o tipo de clientes que é esperado que utilizem o produto. No âmbito do projeto, os clientes esperados são pequenas e médias empresas que necessitem de automatizar processos de negócio e indivíduos que pretendam automatizar o seu trabalho ou vida pessoal;

Recursos Chave: Apresenta o objetivo de listar os recursos essenciais para a concretização do projeto. No âmbito deste projeto, os recursos necessários são a base do projeto *fair-code* n8n.io, assim como o conhecimento tecnológico adjacente;

Canais: Este bloco representa a forma como será entregue o produto ao cliente. No âmbito do projeto, os canais a serem utilizados serão os canais de comunicação do produto E-goi e o *blog* E-goi para manter o cliente sempre atualizado e informado;

Fonte de Retorno: Representa o retorno financeiro e/ou de valor de um produto. No âmbito do projeto, a plataforma trará um maior uso do produto E-goi;

Custos: Este bloco tem como objetivo identificar os custos envolvidos na implementação da ideia. No âmbito do projeto, os custos envolvidos são a necessidade de *hardware*, como servidores, e equipas de desenvolvimento;

Proposta de valor: Representa os motivos para o cliente utilizar e escolher o produto. No âmbito do projeto, a proposta de valor consiste na plataforma de automação de *workflows* de integração que apresenta uma interface gráfica simples e intuitiva que permita ao utilizador ter uma experiência agradável;

3.3 Sumário

Neste Capítulo foram definidos os requisitos da solução. Foi também feita uma análise de valor da solução onde foi descrito o processo de inovação que contempla a identificação da oportunidade, a análise da mesma, a geração de ideias e a seleção da melhor ideia segundo o método AHP. Foi identificado o valor do conceito para o cliente e realizada a proposta de valor. Por último, foi realizado e analisado o modelo Canvas.

Capítulo 4

Desenho

Este capítulo tem como foco o desenho da solução a ser desenvolvida segundo os princípios de software.

4.1 Desenho Arquitetural

Nesta sub secção é realizada uma análise sobre a abordagem arquitetural para o projeto, bem como, algumas alternativas para se apresentar um desenho arquitetural completo. Foram desenvolvidos os diferentes artefactos que servirão de base para a implementação da solução.

A descrição da análise e desenho foi auxiliada pelo Modelo 4+1. A arquitetura de software lida com abstração, decomposição e composição, estilo e estética. Este modelo é composto por cinco vistas ou perspectivas, que permitem abordar separadamente as preocupações de várias partes interessadas da arquitetura. As cinco vistas são projetadas usando um processo de desenvolvimento iterativo centrado na arquitetura e orientado por cenário [43].

Vista lógica: A vista lógica suporta os requisitos funcionais, isto é, o que o sistema deve oferecer em termos de serviços aos utilizadores. O sistema é decomposto num conjunto de abstrações, retiradas do domínio do problema, em forma de classes [43].

Vista de implementação: A vista de implementação foca-se na organização dos módulos do software no ambiente de desenvolvimento. O software é agrupado em pequenos módulos ou subsistemas. Estes são organizados em hierarquias de camadas. Cada camada oferece uma interface bem definida para as camadas abaixo. [43]

Vista de implantação: A vista de implantação foca-se primeiramente nos requisitos não funcionais do sistema como a disponibilidade, confiabilidade, desempenho e escalabilidade. São identificados vários componentes físicos do sistema [43]. No âmbito deste projeto foi desenvolvido um diagrama de implantação.

Vista de processos: A vista de processos tem em consideração alguns requisitos não funcionais como o desempenho e a disponibilidade. Aborda problemas de concorrência e distribuição, integridade do sistema e as principais abstrações da vista lógica que se enquadram na arquitetura de processos [43].

Vista de cenários: A vista de cenários descreve os casos de uso ou cenários que representam a quinta vista. Os cenários descrevem sequências de interações entre objetos e processos. As outras quatro vistas trabalham juntas usando um pequeno conjunto de importantes cenários. No âmbito deste projeto a descrição dos casos de uso é feita no Capítulo 3 e na Secção 3.1.1.

4.1.1 Domínio

O domínio de uma aplicação consiste na definição de todos os conceitos e entidades relacionados com o problema que se pretende resolver. Na Figura 4.1 está representado o modelo de domínio do problema desta dissertação.

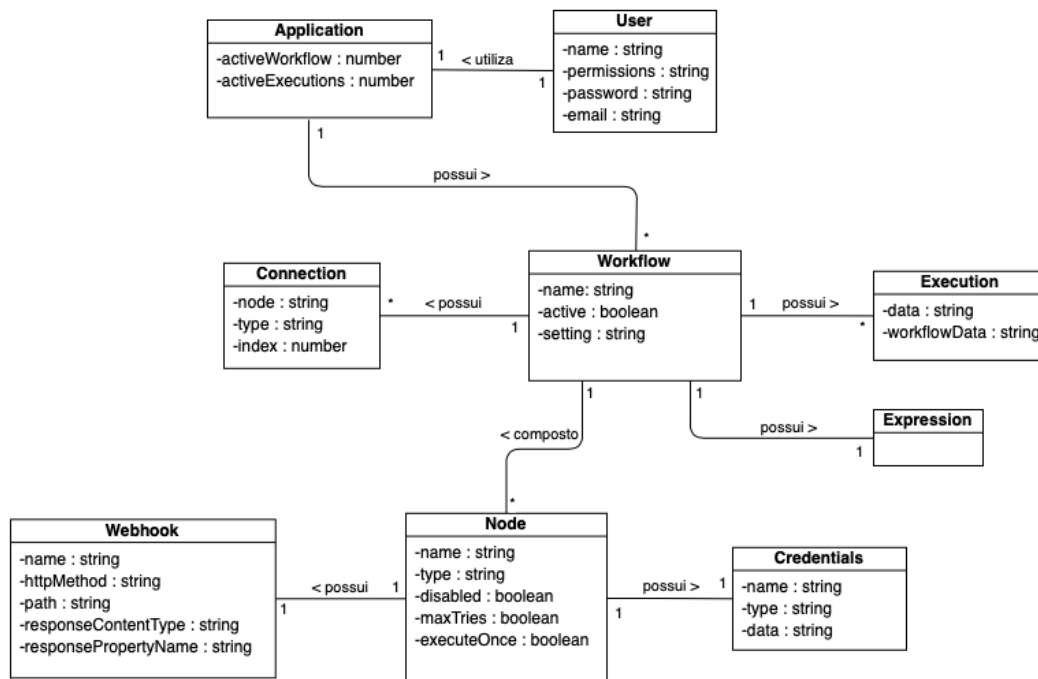


Figura 4.1: Modelo de Domínio

O modelo de domínio apresentado utiliza classes conceptuais do domínio do projeto n8n com algumas alterações para colmatar problemas relativos ao projeto em questão. As classes conceptuais representadas no domínio são:

User: Representa os utilizadores da plataforma e apresenta nome, permissões, username, password e email. Possui um conjunto de credenciais e utiliza uma aplicação.

Credentials: Representa as chaves de API de diversos *nodes* e estão associadas a cada *node*.

Application: Representa a aplicação geral que possui diversos *workflows* e está associada a um utilizador.

Workflow: Representa o fluxo com as conexões entre as diversas aplicações e serviços. Possui conexões entre os *nodes*, um conjunto de execuções, e expressões para o auxílio no tratamento dos dados. É composto por vários *nodes* que representam aplicações e serviços.

Connection: Representa as conexões entre dois *nodes*.

Execution: Representa as diversas execuções de um *workflow*.

Expression: Representa as expressões utilizadas para tratamento de dados dos *workflows*.

Node: Representa as aplicações e serviços que compõem os *workflows* e é composta por vários *webhooks*.

Webhook: Representa um recurso para que cada *node* possa trocar dados em tempo real com outro *node*.

4.1.2 Definição das alternativas

Esta subsecção tem como objetivo apresentar e definir as alternativas de arquitetura para o sistema descrito ao longo deste documento.

Como foi descrito ao longo da Secção 3.2, a implementação de uma plataforma no Goidini, reestruturando o projeto fair-code n8n.io e utilizando o sistema de gestão de utilizadores do Goidini, é a melhor abordagem para a solução do problema.

O projeto n8n.io apresenta código aberto e está disponível para *self-host*. No entanto, a sua arquitetura foi desenvolvida para haver apenas uma única instância para um único utilizador. Tendo por base esta premissa foram estudadas, discutidas e elaboradas duas alternativas de arquitetura para a solução desejada. Definem-se as alternativas como:

Alternativa 1 : Instanciar o projeto n8n.io num servidor dedicado, criando uma integração com o Goidini e a gestão de utilizadores.

Alternativa 2 : Existência de uma instância do projeto n8n.io para cada utilizador.

4.1.3 Vista de Implementação

A vista de implementação ilustra o sistema do ponto de vista do programador. É possível ser representada a partir de um diagrama de componentes.

Alternativa 1

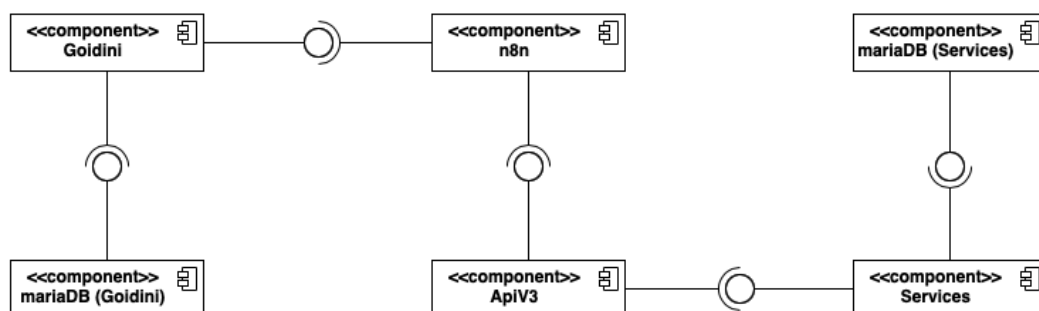


Figura 4.2: Alternativa 1 de Diagrama de Componentes

Através da análise da figura 4.2, verifica-se a existência de 6 componentes:

Goidini: Este componente foi desenvolvido na linguagem de programação Hypertext Preprocessor (PHP) e contém todas as integrações que existem disponíveis com o E-goi.

MariaDB (Goidini): Responsável pela persistência dos dados provenientes do componente Goidini. Este componente é desenvolvido usando a tecnologia MariaDB, um

sistema de gestão de bases de dados relacional com base em MySQL. Aqui também serão guardados os dados provenientes do componente n8n.

n8n: Projeto *fair-code* e *node-oriented* desenvolvido na linguagem de programação TypeScript. Comunica com o componente ApiV3 através do protocolo Hyper Text Transfer Protocol Secure (Hyper Text Transfer Protocol Secure (HTTPS)).

ApiV3: Este componente trata-se de uma Representational State Transfer (Representational State Transfer (REST)) Application Program Interface (API), desenvolvida na linguagem de programação PHP. É destinada para desenvolvedores que queiram integrar a E-goi com outras aplicações e projetos. As comunicações com este componente são feitas através do protocolo Hyper Text Transfer Protocol Secure (HTTPS) e os dados são recebidos como objetos JavaScript Object Notation (JSON).

MariaDB (Services): Responsável pela persistência dos dados provenientes do componente Services. Este componente é desenvolvido usando a tecnologia MariaDB, um sistema de gestão de bases de dados relacional com base em MySQL.

Services: Este componente trata-se de uma Representational State Transfer (REST) Application Program Interface (API) desenvolvida na linguagem de programação PHP. Contém grande parte da lógica de negócio da E-goi. A comunicação entre este componente e o componente ApiV3 é feita através do protocolo Hyper Text Transfer Protocol Secure (HTTPS).

Alternativa 2

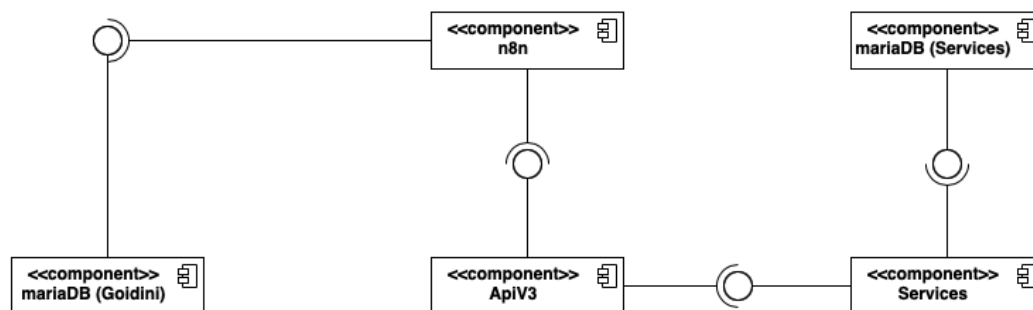


Figura 4.3: Alternativa 2 de Diagrama de Componentes

Através da análise da figura 4.3 verifica-se a existência dos mesmos componentes da Alternativa 1 com a exceção do componente Goidini.

Relativamente às duas alternativas apresentadas, a sua grande diferença é a utilização do componente Goidini, uma vez que a Alternativa 1 se refere à integração do n8 com a gestão de utilizadores do Goidini. Assim, é necessário que estes dois componentes comuniquem de forma a passar dados referentes aos utilizadores, apresentando assim os seus *workflows* e dados do n8n. Na Alternativa 2, o Goidini não assume qualquer papel uma vez que um utilizador terá a sua própria instância no n8n atribuída, sendo apenas necessária a comunicação com a base de dados para guardar os dados dos *workflows*.

4.1.4 Vista de Implantação

A implantação de um projeto consiste na sua instalação nas máquinas, de modo a fornecer o sistema aos utilizadores. Esta etapa apresenta um enorme valor para as organizações, uma vez que as máquinas necessárias para a instalação de *software* implicam um custo elevado.

Um aspeto a ter em conta no desenvolvimento de uma arquitetura de implantação é a gestão das máquinas a serem utilizadas e as instâncias dos diversos componentes, de forma a garantir que não são utilizados recursos desnecessários. Assim, introduz-se o conceito de escalabilidade.

A escalabilidade de uma arquitetura deve ser vista no contexto dos requisitos colocados. Representa a eficácia de cumprir requisitos de capacidade num intervalo de tempo, enquanto continua a satisfazer todos os outros requisitos: funcionais, estatísticos, qualidade de serviço, custo unitário de propriedade, etc. [44]. Existem dois conceitos bem definidos: escalabilidade horizontal e escalabilidade vertical.

A escalabilidade vertical significa adicionar mais recursos à máquina, como RAM e CPU. Por outro lado, a escalabilidade horizontal significa adicionar mais máquinas ao conjunto de recursos. A grande diferença é que a escalabilidade horizontal requer quebrar a lógica sequencial em pequenas partes de forma a serem executadas em paralelo entre múltiplas máquinas. Na escalabilidade vertical a lógica mantém-se inalterada. [45]

Alternativa 1

A primeira alternativa, Figura 4.4, consiste na implementação de uma instância do n8n num servidor dedicado, oferecendo maior independência e recursos. A plataforma comunica com o módulo de gestão de utilizadores do Goidini, que, por sua vez, comunica com a base de dados. A instância do n8n comunica também com o componente da Apiv3 de forma trocar informações relativas às listas e contactos.

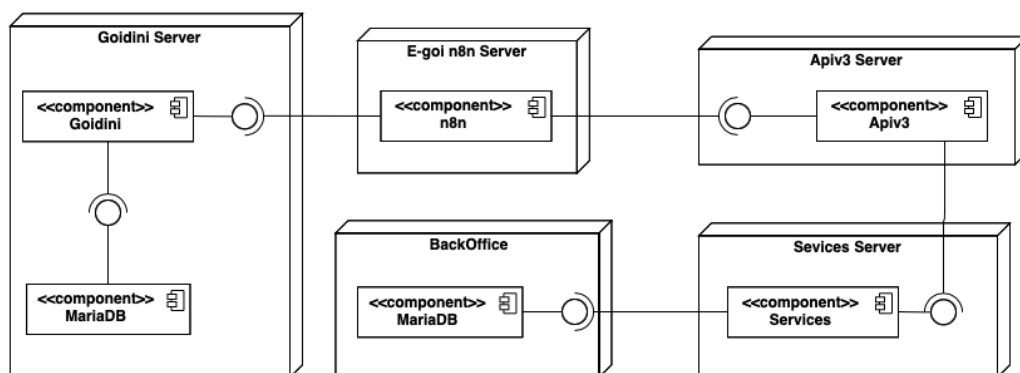


Figura 4.4: Alternativa 1 de Diagrama de Implantação

Esta alternativa segue uma escalabilidade vertical, pois se o número de utilizadores aumentar, aumenta-se a potência da máquina que contém a instância do n8n, permitindo o bom funcionamento da plataforma. Esta alternativa responde ao requisito não funcional de Suportabilidade referido na Subsecção 3.1.2.

Alternativa 2

A segunda alternativa, Figura 4.5, consiste na implementação de múltiplas instâncias numa máquina comum. Esta alternativa de arquitetura visa responder à definição da Alternativa 2 mencionada na Subsecção 4.1.6 do presente Capítulo.

Esta alternativa tem por base o conceito de escalabilidade horizontal, sendo que à medida que aumenta o número de utilizadores, aumenta-se o número de máquinas necessárias para suportar as diversas instâncias do n8n. No entanto, esta escalabilidade traz custos acrescidos à organização, tanto a nível de recursos humanos como de desenvolvimento.

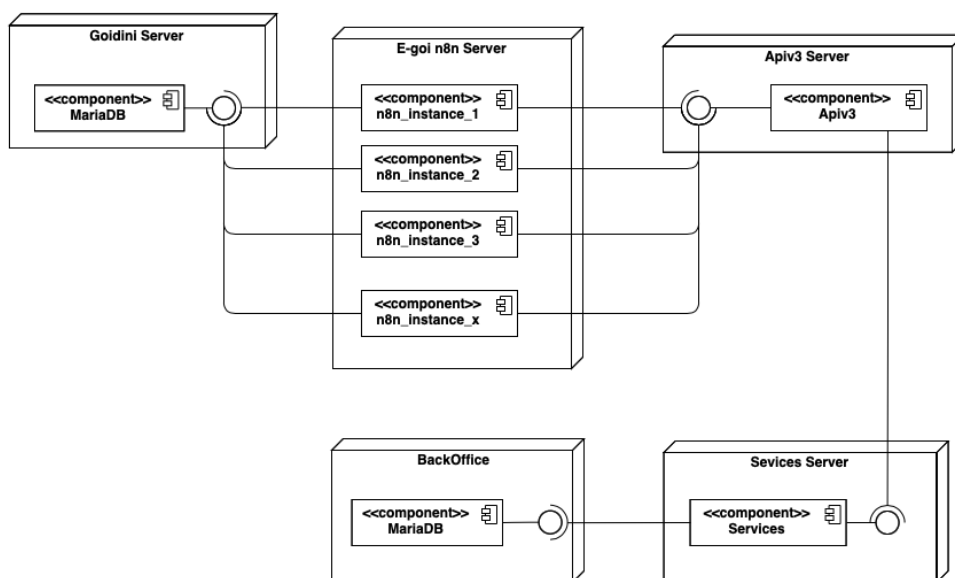


Figura 4.5: Alternativa 2 de Diagrama de Implantação

4.1.5 Modelo de Dados

Nesta subsecção é descrita a base de dados usada pelo projeto n8n.io. O modelo apresentado apresenta apenas as entidades relevantes para o projeto.

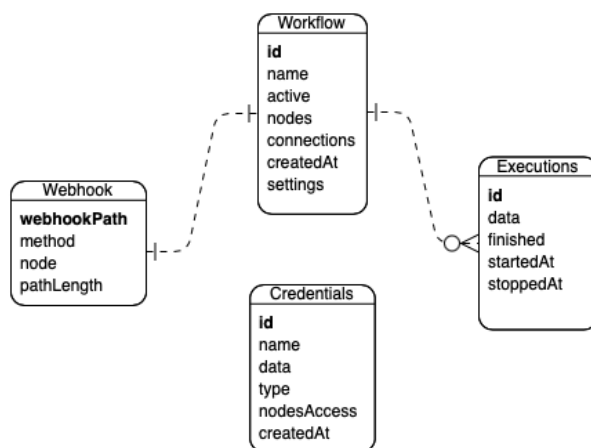


Figura 4.6: Modelo de dados

Segue uma breve descrição das tabelas apresentadas no modelo de dados:

Workflow: Esta tabela contém toda a informação de um workflow. Está relacionada com as tabelas de Webhook e Execution, uma vez que se trata da entidade principal do projeto.

Webhook: Persiste as informações dos webhooks associados a um determinado *node*. Está diretamente relacionado com a entidade Workflow.

Executions: Persiste todas as informações relativas a execuções de um Workflow. Está diretamente relacionado com a entidade Workflow.

Credentials: Esta tabela guarda as credenciais relativas a um determinado *node*. Não está relacionada com nenhuma entidade supramencionada.

Importante referenciar que os *nodes* não são persistidos na base de dados, uma vez que o n8n apresenta uma interface que permite a criação de novos *nodes*. A criação de um novo *node*, será explicada no Capítulo 5 Secção 5.2. Os *nodes* e as suas configurações estão presentes dentro do projeto, e o n8n prepara a sua utilização no momento que o projeto é instanciado inicialmente.

4.1.6 Escolha das Alternativas

A escolha das alternativas de arquitetura do sistema passou por um processo de análise e discussão entre os colaboradores da E-goi e membros da equipa de integrações.

No decorrer da análise e investigação, foi encontrada num fórum da comunidade do n8n, uma informação sobre o lançamento da primeira versão (versão 1.0.0) do n8n.io. Nessa versão será contemplada a gestão de utilizadores. Esta informação foi posteriormente confirmada através do contacto direto com membros da equipa n8n e mudou o paradigma de desenvolvimento do sistema, levando a um processo de investigação mais intenso.

Após um maior conhecimento da ferramenta e da sua arquitetura, verificou-se que existe uma incompatibilidade tecnológica para integrar o sistema de gestão de utilizadores do goidini com o n8n e, um grande esforço no desenvolvimento e criação da estrutura de gestão de utilizadores no n8n, uma vez que o projeto foi inicialmente concebido para ser *self-hosted* e cada instância ser utilizada para um único utilizador.

Tendo por base as informações obtidas relativamente à gestão de utilizadores mencionada anteriormente, foi decidido a utilização do projeto n8n.io seguindo a arquitetura da Alternativa 2, mencionada na subsecção 4.1.6.

Sabendo que apresenta um custo elevado na escalabilidade, foi decidido utilizar a arquitetura da Alternativa para 2 instâncias de uso interno (para o uso de colaboradores da E-goi) e para 8 clientes parceiros da E-goi, uma vez que, numa fase inicial, é possível albergar todas as instâncias apenas num servidor. Com esta alternativa, aumenta-se o conhecimento da ferramenta e obtém-se informações suficientes para melhorar o processo de integração desta ferramenta com a E-goi. O *feedback* tanto dos colaboradores como dos clientes parceiros será importante para futuros desenvolvimentos.

Apesar da mudança e da escolha da arquitetura acima mencionada, a criação do *node* E-goi mantém-se executável e fundamental para a integração do n8n com o E-goi.

4.1.7 Vista de Processos

Nesta subsecção será apresentado um processo que exemplificar a comunicação entre os componentes apresentados na Secção 4.1.3. Como mencionado na secção anterior, o processo de configuração do node E-goi, seguirá a arquitetura da Alternativa 2.

Configurar Node E-goi

Para perceber o processo de configuração do *node* E-goi, utilizou-se um diagrama de sequência.

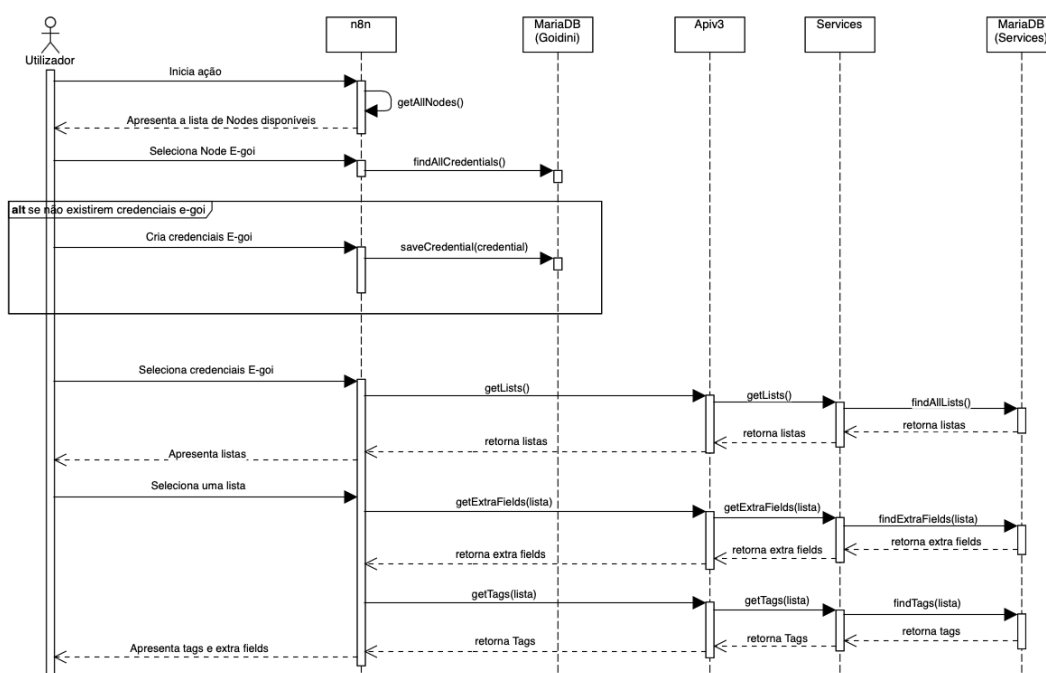


Figura 4.7: Diagrama de sequência do UC03

Como é possível perceber através da análise do diagrama de sequência apresentado na figura 4.7, o utilizador inicia a ação de configuração do *node* E-goi, selecionando uma opção para visualizar todos os *nodes* disponíveis. O n8n obtém os *nodes*. O n8n não necessita recorrer a uma procura na base de dados para obter os *nodes*, como mencionado na Subsecção 4.1.5. O n8n apresenta a lista de *nodes* e o utilizador procede à escolha do *node* E-goi. Após a escolha, o n8n verifica se existem credenciais para o *node* E-goi guardadas. Caso não existam, o utilizador deverá proceder à sua criação. Após a inserção das credenciais, e caso sejam válidas, estas serão armazenadas na base de dados. O utilizador escolhe a credencial e o n8n comunica com a APIv3 através do protocolo HTTPS, para obter todas as listas de contactos disponíveis associadas à credencial selecionada. Após essa procura, o n8n apresenta as listas de contactos e o utilizador seleciona a lista de contactos que pretende que seja utilizada para a criação de um novo contacto ou atualização de um contacto já existente. A seleção da lista pretendida, permite a obtenção dos campos extra e das *tags* associadas a essa lista.

Seguindo este processo, o utilizador pode adicionar as expressões e os campos que quer modificar, uma vez que apresenta o *node* configurado e pronto para ser conectado com

múltiplos *nodes*, de forma a passar os dados relevantes para a criação de um *workflow* funcional.

4.2 Sumário

Neste capítulo foi apresentado o desenho da solução recorrendo a boas práticas de arquitetura de software. Foram definidas alternativas de arquitetura e apresentada a vertente técnica das mesmas. A alternativa escolhida foi justificada devidamente.

Segue-se a implementação da solução escolhida, tendo por base o planeamento efetuado no decorrer deste capítulo.

Capítulo 5

Implementação

A análise de um sistema resulta na sua implementação. Neste capítulo é apresentada a implementação da solução encontrada e explicada nos capítulos anteriores. Como a criação do *node* E-goí é a principal etapa de integração do n8n com a E-goí, é apresentada a metodologia de criação de novos *nodes*, de forma a ter uma total compreensão do processo realizado. É também apresentada a criação das instâncias n8n, modificação da interface gráfica e a criação das tabelas de base de dados.

5.1 Metodologia de Criação de Novos Nodes

Para permitir uma maior panóplia de integrações, o n8n apresenta uma metodologia estandarizada, que permite a desenvolvedores criar novos *nodes* com relativa facilidade.

Para uma melhor compreensão, apresenta-se um diagrama de pacotes do n8n.

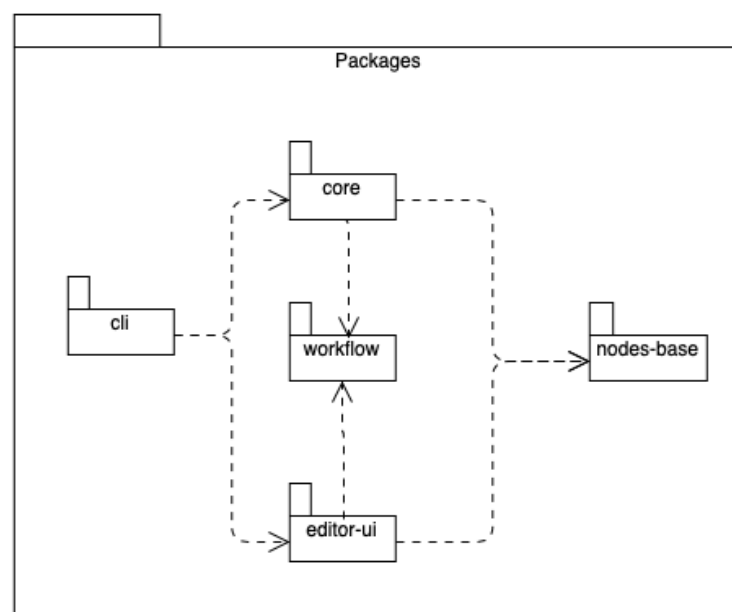


Figura 5.1: Diagrama de Pacotes

O n8n está dividido em diferentes módulos, que fazem todos parte de um *monorepo*. Os principais módulos estão representados na figura 5.1:

packages: Pacote onde estão os diferentes módulos que compõem o n8n.

cli: Módulo que contém o código responsável por correr o *frontend* e *backend*.

core: Um dos módulos principais que contém o código principal que trata da execução e ativação de *workflows* e *webhooks*.

workflow: Módulo onde está presente o código responsável pela criação de *workflows*. Apresenta as interfaces que são utilizadas pelo *frontend* e *backend*.

editor-ui: Este módulo é composto pelos componentes do *frontend*.

nodes-base: Módulo que contém todos os *nodes* do n8n. É neste módulo que são criados novos *nodes*.

Tendo uma visão geral dos módulos do n8n, procede-se à explicação do processo de criação de novos *nodes*. Para iniciar o processo, é necessário que se clone o repositório do projeto que se encontra disponível no GitHub do n8n. Para tal é utilizada a ferramenta Git.

Para a criação dos *nodes*, é necessário trabalhar em 2 módulos dentro do módulo *nodes-base* e um ficheiro de configuração. Estes são:

nodes: Este módulo contém as configurações de todos os *nodes* do n8n.

credentials: Contém as credenciais que os diferentes *nodes* usam. Cada *node* pode optar usar *OAuth2* ou *API Key*.

package.json: Ficheiro que contém todos os pacotes *npm* que os *nodes* utilizam. Contém todas as credenciais e o *nodes* que são carregados quando é inicializado.

Dentro do módulo *nodes-base/nodes*, é criado um diretório para o novo *node* que se pretende criar. Dentro desse diretório, coloca-se o *icon* a ser utilizado, e um ficheiro TypeScript onde são definidas todas as configurações e propriedades do novo *node*.

Uma vez criada a estrutura de configuração do *node*, é necessário criar a estrutura para as credenciais. No módulo *nodes-base/credentials*, é inserido um ficheiro TypeScript relativo às credencias do novo *node*. Nesse ficheiro é especificado que tipo de credenciais é utilizado.

Assim que todo o desenvolvimento esteja concluído, procede-se à submissão do novo *node*. A submissão é feita através de um *merge request* para validação. Após a validação, o *node* é lançado no repositório.

5.2 Criação Node E-goi

A criação do *node* E-goi segue a metodologia apresentada na subsecção anterior. Desta forma, o desenvolvimento iniciou-se com o clone do repositório do projeto. Após terem sido instaladas as dependências necessárias, procedeu-se à criação da estrutura para o novo *node*.

Inicialmente, foi criado dentro do módulo *nodes-base/nodes*, o diretório com os ficheiros para a criação do *node*. O diretório apresenta os seguintes ficheiros:

Egoi.node.json: Ficheiro onde são colocadas as informações relativas ao *node* E-goi. Estas informações são a versão e endereços eletrónicos relativos à documentação do *node*.

Egoi.node.ts: Ficheiro que contém a classe principal do novo *node* E-goi.

ego.png: Logótipo da E-goi com dimensão fixa de 60x60 pixels.

GenericFunctions.ts: Ficheiro que contém funções auxiliares usadas na implementação do *node*.

Interfaces.ts: Ficheiro onde estão definidas interfaces usadas na classe principal.

No ficheiro *Egoi.node.ts*, inicia-se o desenvolvimento, com a criação da classe principal que implementa uma interface denominada *INodeType* que é utilizada para a criação de novos *nodes*. As propriedades dessa interface são também interfaces, para se manter uma estrutura padronizada que facilite a criação de novas integrações. De forma a manter uma boa percepção do desenvolvimento realizado, apenas serão mencionadas as interfaces utilizadas na criação do *node* E-goi.

Foram utilizadas 2 propriedades e um método que estão definidos na interface *INodeType*. As propriedades utilizadas foram a *descriptions* e *methods* e o método utilizado foi o *execute*. Procede-se a explicação e composição dos mesmos.

Propriedade description

Esta propriedade apresenta a interface *INodeTypeDescription* como tipo. Com esta interface, define-se as informações que serão utilizadas no *frontend* como o nome do *node*, o *icon*, as cores e os parâmetros. São definidos os parâmetros necessários para tornar o *node* funcional. Os principais parâmetros criados são:

Resource: Define o recurso onde a operação definida vai operar. Foi definido apenas o recurso contacto.

```
1   properties: [  
2     {  
3       displayName: 'Resource',  
4       name: 'resource',  
5       type: 'options',  
6       required: true,  
7       options: [  
8         {  
9           name: 'Contact',  
10          value: 'contact',  
11        },  
12      ],  
13      default: 'contact',  
14      description: 'The resource to operate on.',  
15    },  
16    ...
```

Listing 5.1: Parâmetro resource

Operation: Define a operação que será executada ao recurso selecionado. Definiram-se 4 operações: criar contacto, atualizar contacto, listar informações de um contacto e listar todos os contactos.

```
1 ...  
2 {  
3   displayName: 'Operation',  
4   name: 'operation',  
5   type: 'options',  
6   required: true,  
7   options: [  
8     {  
9       name: 'Create',  
10      value: 'create',  
11      description: 'Create a member',  
12    },  
13    {  
14      name: 'Get',  
15      value: 'get',  
16      description: 'Get a member',  
17    },  
18    {  
19      name: 'Get All',  
20      value: 'getAll',  
21      description: 'Get all members',  
22    },  
23    {  
24      name: 'Update',  
25      value: 'update',  
26      description: 'Update a member',  
27    },  
28  ],  
29  default: 'create',  
30  description: 'The operation to perform.',  
31 },  
32 ...
```

Listing 5.2: Parâmetro operation

ListId: Este parâmetro define a lista ao qual será executada a operação do contacto. Se a operação escolhida for criar contacto, então um contacto será adicionado à lista que tiver sido escolhida.

Define-se como tipo *options* e adiciona-se uma opção que permite carregar dados chamando um método definido na propriedade *methods*.

```
1 ...  
2 {  
3   displayName: 'List ID',  
4   name: 'list',  
5   type: 'options',  
6   typeOptions: {  
7     loadOptionsMethod: 'getLists',  
8   },  
9   ...
```

Listing 5.3: Exemplo de loadOptionsMethod

Para cada operação escolhida, é possível definir que parâmetros são mostrados no *frontend*. Por exemplo, se apenas se pretender apresentar um parâmetro de *Email* para a criação de um contacto, então define-se a seguinte opção:

```
1 ...  
2 {  
3   displayName: 'Email',  
4   name: 'email',  
5   type: 'string',  
6   displayOptions: {  
7     show: {  
8       operation: [  
9         'create',  
10      ],  
11    },  
12  },  
13  ...
```

Listing 5.4: Exemplo de displayOptions

Propriedade methods

Esta propriedade define métodos a serem usados pelos parâmetros criados na propriedade *description*. Foram definidos os métodos *getLists*, *getExtraFields* e *getListTags*. Estes métodos permitem a obtenção de todas as listas, dos campos extra de uma lista e das *tags* de uma lista, respetivamente. Os dados são obtidos com um pedido à Apiv3 do E-goi. A função que realiza esses pedidos está definida no ficheiro *GenericFunction.ts*, e o código está disponível no Anexo B.1.

Método execute

Este método é executado sempre que o *node* for executado. Com este método, tem-se acesso aos dados de entrada e aos parâmetros inseridos pelo utilizador. Com a obtenção dos parâmetros, é possível verificar qual é a operação a ser realizada. Para exemplificar, o processo para a adição de um novo utilizador, é o seguinte:

1. Obtenção dos parâmetros referentes à operação *create*;
2. Transformação de dados para coincidir com o mesmo tipo de dados da Apiv3;
3. Construir *body* para realizar o pedido;

4. Realizar o pedido;
5. Em caso de sucesso retornar o id do contacto.

O Anexo B.2 representa o código de exemplo, para o mapeamento dos dados na adição de um novo contacto a uma lista.

5.2.1 Configurar Credenciais

Tendo criado a estrutura do *node*, é necessário criar a estrutura que permite a inserção da autenticação para a comunicação com a Apiv3. Deste modo, dentro do módulo *nodes-base/credentials*, foi criado o ficheiro *EgoiApi.credentials.ts*. Como a Apiv3 usa o mecanismo de autenticação por API key, define-se o tipo de credencial como especificado no código 5.6.

```
1 import {
2   ICredentialType,
3   INodeProperties,
4 } from 'n8n-workflow';
5
6
7 export class EgoiApi implements ICredentialType {
8   name = 'egoApi';
9   displayName = 'E-ego API';
10  documentationUrl = 'ego';
11  properties: INodeProperties[] = [
12    {
13      displayName: 'API Key',
14      name: 'apiKey',
15      type: 'string',
16      default: '',
17    },
18  ];
19 }
20 }
```

Listing 5.5: Ficheiro *EgoiApi.credentials.ts*

Após a criação do ficheiro, adiciona-se o caminho relativo do novo ficheiro das credenciais no ficheiro *package.json*, de forma a registar as credenciais.

Por fim, acrescenta-se o parâmetro *credentials* à propriedade *description* do ficheiro *Egoi.node.ts*.

```
1 ...
2 credentials: [
3   {
4     name: 'egoApi',
5     required: true,
6   },
7 ],
8 ...
```

Listing 5.6: Ficheiro *EgoiApi.credentials.ts*

5.2.2 Publicação do node

Após o desenvolvimento do novo *node* E-goi e da validação concluída, este ficou disponível para utilização na plataforma n8n.

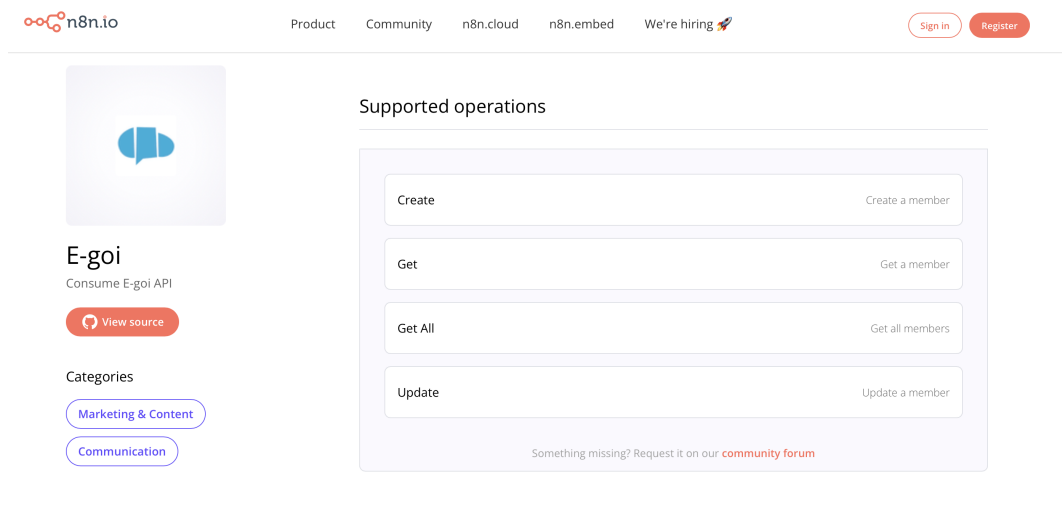


Figura 5.2: Publicação do node E-goi

Esta primeira etapa, é de grande importância para a integração do E-goi com a plataforma. Desta forma, não só é possível a utilização do *node* E-goi internamente, como é possível a sua utilização para quem utilizar o n8n.io.

5.3 Interface Gráfica

Para uma melhor compreensão do trabalho realizado na subsecção anterior, são apresentadas as modificações gráficas relativas à criação do novo *node* E-goi e à interface gráfica do ambiente de criação de *nodes*.

Na figura 5.3 está representado o sistema n8n com alterações a nível de cor e *branding* ajustados à imagem do E-goi.

A interface gráfica é composta por 3 componentes principais:

NodeView (representado a verde) Este módulo contém a parte principal do sistema, isto é, a área de criação de *workflows*. Representado a vermelho está o componente que permite adicionar novos *nodes* à área de criação. Nesta área, é possível adicionar, configurar, conectar diversos tipos de *nodes*, de forma a criar os mais diversos tipos de *workflows*.

MainHeader (representado a azul) Este módulo contém o nome do *workflow* que está a ser trabalhado, opção de guardar e ativação do mesmo.

MainSidebar (representado a laranja) Este módulo contém o *branding* e as barras laterais que permitem a criação e visualização de *workflows*, credenciais e execuções.

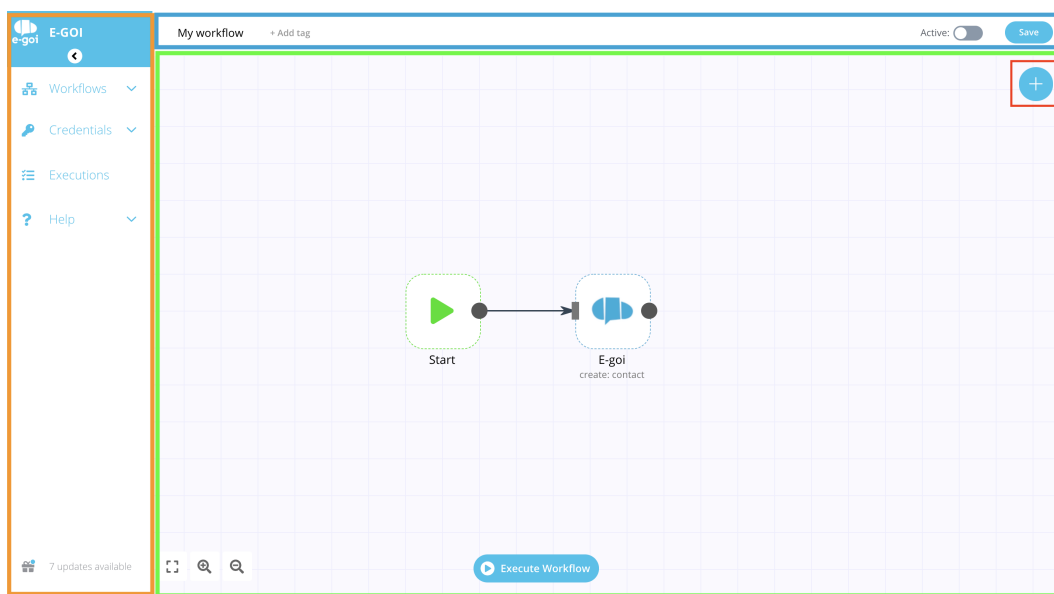


Figura 5.3: Área de criação de Workflows

Relativamente ao *node* E-goi, a sua criação resultou na interface gráfica representada na figura 5.4.

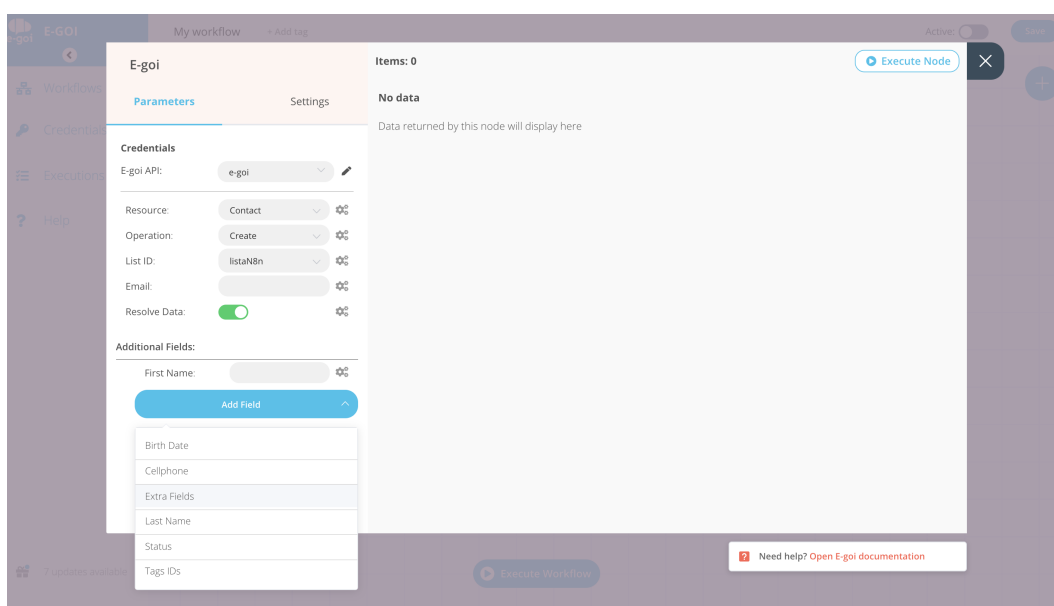


Figura 5.4: Área de configuração do node E-goi

Verifica-se a existência dos parâmetros que foram criados e explicados na Subsecção 5.2. O processo realizado nesta área está representado no caso de uso UC definido no Capítulo 3 Subsecção 3.1.1 e no diagrama de sequência representado na figura 4.7 referente à vista de processos do Capítulo 4.

5.4 Instâncias

Como referido no Capítulo 4 Subsecção 4.1.6, foi decidida a criação de 10 instâncias n8n num servidor dedicado, sendo 2 instâncias para uso interno e 8 instâncias para clientes parceiros. De forma a proteger informações sensíveis relativas à criação das instâncias, serão usados dados fictícios relativos a uma instância.

Para o bom funcionamento, cada instância necessita de aproximadamente de 300mb de RAM. Assim, foi criado um servidor com capacidade de correr as 10 instâncias em simultâneo, sem grande problemas de performance.

A inicialização de uma instância procedeu-se através da criação e execução de um ficheiro `docker-compose.yml`. É pretendido que o acesso se faça através de um domínio específico com um *subpath* dedicado a cada instância. Assim, o acesso para uma instância com um subdomínio `conta1`, deveria aceder à sua instância através de `https://n8negoidomain.com/conta1`. O acesso a cada instância é protegido com um sistema de OAuth2. O *username* e *password* são definidos num ficheiro de configuração, como está exemplificado no Anexo B.4.

O ficheiro `docker-compose.yml` terá as configurações necessárias para executar os projetos que foram colocados no servidor. As configurações mais relevantes serão as variáveis de ambiente necessárias para conectar com a base de dados e para definir o domínio. Um exemplo do ficheiro está presente no Anexo B.5.

5.5 Base de Dados

A base de dados escolhida para persistir os dados, foi a base de dados do Goidini, ou seja, uma base de dados em MariaDB. Para cada utilizador, foram criadas as tabelas para guardar as credenciais, tabelas, *workflows* e *webhooks*.

Por se tratar de uma base de dados MariaDB, a definição das tabelas foi realizada com recurso a um subconjunto de instruções Structured Query Language (SQL). O subconjunto em questão foi o *Data Definition Language* (DDL), que é geralmente utilizado para criar e gerir a estrutura de uma base de dados. Os comandos subjacentes a este subconjunto são:

CREATE Utilizado para criar objetos numa base de dados.

ALTER Utilizado para alterar a estrutura de um objeto.

DROP Utilizado para remover um objeto de uma base de dados.

Para a criação das tabelas necessárias, foi utilizada a declaração **CREATE**, como está representado no excerto 5.7. A criação das tabelas segue a estrutura apresentada na Subsecção 4.1.5.

Para o n8n saber qual a tabela onde tem de guardar os dados, é definida uma constante no ficheiro `docke-compose.yml`, como exemplificado no ficheiro presente no Anexo B.4. É definida a constante `DB_TABLE_PREFIX` com o prefixo dado na criação das tabelas. No exemplo do excerto, a constante assume o valor de `conta1`.

```
1
2 CREATE TABLE IF NOT EXISTS conta1_webhook_entity (
    workflowId int NOT NULL, webhookPath varchar(255) NOT
    NULL, method varchar(255) NOT NULL, node varchar(255)
    NOT NULL, PRIMARY KEY (webhookPath, method)) ENGINE=
    InnoDB;
3
4 CREATE TABLE IF NOT EXISTS conta1_credentials_entity (id
    int NOT NULL AUTO_INCREMENT, name varchar(128) NOT NULL
    , data text NOT NULL, type varchar(32) NOT NULL,
    nodesAccess json NOT NULL, createdAt datetime NOT NULL,
    updatedAt datetime NOT NULL, PRIMARY KEY (id)) ENGINE=
    InnoDB;
5
6 CREATE TABLE IF NOT EXISTS conta1_execution_entity (id int
    NOT NULL AUTO_INCREMENT, data text NOT NULL, finished
    tinyint NOT NULL, mode varchar(255) NOT NULL, retryOf
    varchar(255) NULL, retrySuccessId varchar(255) NULL,
    startedAt datetime NOT NULL, stoppedAt datetime NOT
    NULL, workflowData json NOT NULL, workflowId varchar
    (255) NULL, PRIMARY KEY (id)) ENGINE=InnoDB;
7
8 CREATE TABLE IF NOT EXISTS conta1_workflow_entity (id int
    NOT NULL AUTO_INCREMENT, name varchar(128) NOT NULL,
    active tinyint NOT NULL, nodes json NOT NULL,
    connections json NOT NULL, createdAt datetime NOT NULL,
    updatedAt datetime NOT NULL, settings json NULL,
    staticData json NULL, PRIMARY KEY (id)) ENGINE=InnoDB;
```

Listing 5.7: Exemplo da definição das tabelas

5.6 Sumário

Neste capítulo foi explicado todo o desenvolvimento da solução. Inicialmente foi apresentada a metodologia de criação de novos *nodes*. Seguindo essa metodologia explicou-se o desenvolvimento do *node* E-goí, seguindo da apresentação das modificações da interface gráfica. Por último foram apresentadas as soluções desenvolvidas relativamente às instâncias do n8n e da base de dados utilizada.

No capítulo seguinte são apresentadas as experiências e avaliações da solução desenvolvida.

Capítulo 6

Experimentação e Avaliação

O desenvolvimento de um produto culmina na sua avaliação, de forma a verificar a qualidade da solução desenvolvida. Neste capítulo é apresentado o processo de avaliação do projeto desenvolvido, bem como a análise aos resultados obtidos.

O primeiro passo para uma boa avaliação, é a definição de métricas e de hipóteses para as avaliar. Uma vez que o paradigma do projeto mudou com a escolha da alternativa arquitetural explicadas no Capítulo 4.1.6, foram formuladas métricas e hipóteses relativas à satisfação do utilizador na utilização da plataforma e na criação de *workflows* com o *node* E-goi.

6.1 Métricas

As métricas consideradas para a avaliação da solução foram:

Métrica 1: Satisfação do utilizador quanto à utilização da plataforma.

Métrica 2: Satisfação do utilizador quanto à criação de *workflows* com o *node* E-goi.

A definição destas métricas tem o objetivo de avaliar o *node* E-goi desenvolvido, quanto às funcionalidades que oferece e às possibilidades de integração. Avaliar a utilização da plataforma é importante, para perceber se é considerada uma mais valia e capaz de ser competitiva no mercado de automação de *workflows* de integrações.

6.2 Hipóteses

Após a definição das métricas, é necessário definir as hipóteses necessárias para as testar. Para cada métrica foram formuladas 2 hipóteses.

Relativamente à **Métrica 1** foram definidas as seguintes hipóteses:

H0: Satisfação do utilizador (utilização da plataforma)

H1: Questionário de satisfação possui um nível superior a 3 numa escala de 1-5.

H2: Questionário de satisfação não possui um nível superior a 3 numa escala de 1-5.

Para a **Métrica 2** foram definidas as seguintes hipóteses:

H0: Satisfação do utilizador (criação de *workflows* com o *node* E-goi.)

H1: Questionário de satisfação possui um nível superior a 3 numa escala de 1-5.

H2: Questionário de satisfação não possui um nível superior a 3 numa escala de 1-5.

Para todas as hipóteses anterior, foi escolhido o valor superior a 3, uma vez que este representa um nível de satisfação neutra. Como a solução desenvolvida, ainda é uma solução que está num ciclo de vida muito inicial, entendeu-se que seria o valor justo para avaliar a solução.

6.3 Metodologia

De forma a avaliar as métricas e hipóteses formuladas, é necessário definir a metodologia a utilizar. Para a avaliação de ambas as métricas, foi utilizado um inquérito de satisfação de modo a medir a usabilidade do sistema. Este inquérito, disponível no Anexo C, está dividido em duas secções. A primeira secção apresenta 6 questões relacionadas com a utilização da plataforma e a segunda secção apresenta 5 questões relacionadas com a criação de *workflows* com o *node* E-goi.

Para a avaliação de cada questão foi utilizada a seguinte escala de classificação:

Tabela 6.1: Escala de classificação das questões

Escala	Descrição
1	Muito insatisfeito
2	Insatisfeito
3	Neutro
4	Satisfeito
5	Muito satisfeito

Este inquérito foi disponibilizado a colaboradores da E-goi que utilizaram a plataforma e aos 8 clientes parceiros.

6.4 Análise de Resultados

Após ser disponibilizado o inquérito aos colaboradores e clientes, obteve-se uma amostra reduzida de 15 elementos. Os resultados das perguntas do inquérito podem ser vistos em forma de gráfico de barras no Anexo D.

Relativamente à **Métrica 1**, apresentam-se os resultados com o valor percentual de respostas para cada escala definida anteriormente.

As questões formuladas nesta secção foram:

Questão 1: Considera o acesso à plataforma prático e intuitivo?

Questão 2: A plataforma permite criar um workflow de forma fácil?

Questão 3: Considera rápida e intuitiva a adição de um node à área de criação de workflows?

Questão 4: A plataforma permite responder às necessidades relativas aos workflows que pretende automatizar?

Questão 5: De 0 a 5 indique o grau de confiança a utilizar a plataforma.

Questão 6: Considera o tempo de resposta das operações tolerável?

Tabela 6.2: Resultados do inquérito de satisfação da utilização da plataforma

Questão	1	2	3	4	5
Questão 1	0%	0%	33,3%	33,3%	33,3%
Questão 2	0%	0%	0%	53,3%	46,7%
Questão 3	0%	0%	0%	26,7%	73,3%
Questão 4	0%	0%	6,7%	46,7%	46,7%
Questão 5	0%	0%	0%	46,7%	53,3%
Questão 6	0%	0%	0%	46,7%	53,3%

Tendo em consideração as hipóteses formuladas para esta métrica e os resultados obtidos, onde a média das respostas é de aproximadamente 4, é possível afirmar que existe uma satisfação global na utilização da plataforma. No entanto, é possível destacar alguns resultados. Relativamente ao acesso da plataforma existe uma satisfação neutra, e é o principal ponto a ter em consideração em desenvolvimentos futuros. Já era expectável que este ponto tivesse uma variação maior e um nível de satisfação menor uma vez que o acesso à plataforma é feito de forma direta e já com uma conta definida sem possibilidade de criar novos utilizadores. É também importante referir que existe uma satisfação elevada, relativamente à facilidade de adição de novos *nodes* à área de criação de *workflows*.

Relativamente à **Métrica 2**, apresentam-se os resultados com o valor percentual de respostas para cada escala definida anteriormente.

Questão 1: Considera de fácil aprendizagem o processo de configuração do node?

Questão 2: Considera o node útil relativamente aos dados que pode enviar para a sua conta E-goi? (ex: adicionar um contacto a uma lista da sua conta E-goi).

Questão 3: A criação de expressões é de fácil aprendizagem?

Questão 4: Considera os workflows com o node E-goi relevantes?

Questão 5: De 0 a 5 indique o grau de confiança com a utilização do node E-goi nos workflows.

Tabela 6.3: Resultados do inquérito de satisfação da utilização da plataforma

Questão	1	2	3	4	5
Questão 1	0%	0%	0%	53,3%	46,7%
Questão 2	0%	0%	0%	33,3%	66,7%
Questão 3	0%	0%	13,3%	66,7%	20%
Questão 4	0%	0%	6,7%	33,3%	66,7%
Questão 5	0%	0%	0%	46,7%	53,3%

Tendo em consideração as hipóteses formuladas para esta métrica e os resultados obtidos, onde a média das respostas é de aproximadamente 4, é possível afirmar que existe uma satisfação global quanto à criação de workflows com o *node* E-goi.

Importante destacar os resultados obtidos nas questões 2 e 4, uma vez que apresentaram um nível de satisfação elevados em dois pontos importantes na integração da E-goi com outras aplicações e serviços.

6.5 Sumário

Neste capítulo são apresentadas as as métricas e hipóteses para avaliar a solução desenvolvida. Através desta avaliação é possível perceber que pontos podem ser melhorados e tidos em consideração em desenvolvimentos futuros.

Após ser validada a solução, é de extrema importância retirar conclusões, bem como definir o trabalho futuro, de modo a melhorar a solução desenvolvida.

Capítulo 7

Conclusão

Neste capítulo é abordado e analisado todo o trabalho realizado, bem como as limitações que foram sentidas no decorrer do projeto e definido o trabalho futuro.

7.1 Objetivos concluídos

O projeto documentado teve como principal objetivo, o desenvolvimento de uma ferramenta robusta e escalável que permita a gestão de *workflows* de integração, baseados em *triggers* e *actions*, de modo a alargar as possibilidades de integrações com o E-goi.

Na fase inicial do projeto, foram definidos objetivos específicos que contribuem para a concretização do objetivo principal. Os objetivos encontram-se definidos na Secção 1.3 e são os seguintes:

1. Identificar problemas relacionados com as integrações na E-goi;
2. Análise do fluxo geralmente utilizado nas integrações;
3. Identificar soluções para os problemas identificados;
4. Definir uma arquitetura para a melhor solução encontrada;
5. Implementar a arquitetura definida;
6. Integrar a solução desenvolvida na plataforma E-goi;
7. Testar a solução.

O objetivo 1 foi atingido através de uma análise à plataforma da E-goi, bem como o testemunho de diversos colaboradores da E-goi. Assim, foi possível identificar os seguintes problemas:

1. Existência de um ambiente limitado que automatize *workflows*;
2. Estratégias de marketing cada vez mais complexas;
3. Demanda de serviços mais ready-to-go.

O objetivo 2 foi atingido, uma vez que foram estudadas diversas plataformas de automação de *workflows*, permitindo obter um maior conhecimento do processo geralmente utilizado em integrações de diversas aplicações. Esse estudo está documentado na Secção 2.4.

O objetivo 3 foi atingido identificando soluções para os problemas encontrados no objetivo 1. As soluções encontradas foram apresentadas, analisadas e foi obtida a melhor solução

através do modelo *New Concept Development*. Todo o processo está documentado na Secção 3.2.

O objetivo 4 foi atingido através do desenvolvimento descrito no Capítulo 4. Neste capítulo foram abordadas diversas alternativas de arquitetura e descritos problemas encontrados nesse processo. No final foi escolhida a melhor alternativa para o desenvolvimento do projeto.

Relativamente ao objetivo 5, a abordagem consistiu na implementação do sistema proposta no Capítulo 4. Assim, considera-se que este objetivo foi atingido.

O objetivo 6 não foi atingido uma vez que seria expectável a existência de um ponto unificado que permitisse a utilização da plataforma. Como o paradigma da arquitetura foi alterado e explicado no Capítulo 4, a existência da solução da plataforma não foi possível de desenvolver.

Por último o objetivo 7 foi atingido recorrendo a testes de usabilidade apresentados no Capítulo 6. Os resultados analisados permitiram identificar a satisfação global da solução desenvolvida e os pontos a melhorar futuramente.

Assim, considera-se que o projeto se revelou bem sucedido, capaz de responder ao objetivo principal inicialmente proposto. Considera-se que a solução encontrada permite alargar as possibilidades de integrações com o E-goi. No entanto, considera-se de extrema importância realçar as limitações sentidas no decorrer do projeto e que proporcionaram o incumprimento do objetivo 6.

7.2 Limitações

O primeiro grande desafio sentido, foi a habituação e conhecimento da plataforma E-goi, uma vez que nunca tinha sido usada e para o desenvolvimento de integrações com a E-goi, é importante conhecer bastante bem o produto e a plataforma.

Relativamente ao desenvolvimento da arquitetura e implementação da solução, foram sentidas algumas limitações. No momento do desenvolvimento da arquitetura da solução, foram introduzidas dificuldades que não eram expectáveis e que mudaram o rumo da solução. A primeira dificuldade foi a identificação da incompatibilidade tecnológica entre o n8n e o Goidini, que não foi identificada numa primeira fase de análise e investigação. Mas, como é natural, com o aumento do conhecimento da plataforma, foi possível perceber essa limitação. Assim, foram descritas as duas alternativas para desenvolver o projeto inicialmente proposto e foi identificada a solução que melhor responde às necessidades da E-goi.

Apesar da solução apresentar bons resultados de usabilidade, do ponto de vista do negócio, torna-se insustentável para ser utilizada para todos os clientes do E-goi, uma vez que, seriam necessárias muitas mais máquinas para correr todas as instâncias necessárias para todos os clientes, trazendo enormes custos à organização.

Assim, concluiu-se que a melhor abordagem seria esperar pela gestão de utilizadores, que será lançada na primeira versão do n8n, uma vez que se trata de um projeto complexo. O desenvolvimento da gestão de utilizadores por parte da equipa dedicada, apresenta uma maior garantia quanto à eficácia e eficiência dessa componente. Com este componente será possível correr o n8n numa única instância no servidor, levando à escolha e desenvolvimento da alternativa 1 apresentada no Capítulo 4.

Com a sucessiva utilização por parte dos colaboradores e dos dos clientes parceiros da solução desenvolvida, é possível obter informações valiosas relativamente ao desenvolvimento futuro e posição estratégica por parte da E-goi.

7.3 Trabalho Futuro

Relativamente ao trabalho futuro, este passa pela preparação, tanto do Goidini como do sistema E-goi, para o lançamento da componente de gestão de utilizadores. Desta forma será possível concluir o objetivo 6.

Para além da preparação arquitetural e técnica, a solução atual vai precisar de uma manutenção contínua, para que sejam suprimidos eventuais erros ou dificuldades inerentes a qualquer sistema.

Bibliografia

- [1] Fabio Casati e Ming-Chien Shan. «Process Automation as the Foundation for E-Business». Em: *Proceedings of the 26th International Conference on Very Large Data Bases*. VLDB '00. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 2000, pp. 688–691. isbn: 1558607153.
- [2] *E-goi - O que fazemos*. Out. de 2020. url: <https://www.e-goi.com/pt/o-que-fazemos/>.
- [3] *Goidini*. url: <https://goidini.e-goi.com/index/home>.
- [4] Anthony Bagshaw. «What is marketing automation?» Em: *Journal of Direct, Data and Digital Marketing Practice* 17.2 (2015), pp. 84–85. doi: 10.1057/ddmp.2015.46.
- [5] Jeffrey O. Grady. «System Integration». Em: (2020). doi: 10.1201/9781003068815.
- [6] Andrzej Cichocki et al. *Workflow and process automation: concepts and technology*. Vol. 432. Springer Science & Business Media, 2012.
- [7] Darrell M. West. *The future of work: robots, AI, and automation*. Brookings Institution Press, 2019.
- [8] Tim Eherns. *Integration*. url: <https://searchcustomerexperience.techtarget.com/definition/integration>.
- [9] Katharine Marais. *Workflow Automation: A Comprehensive Guide, Including Automation Tools, ROI, and Example Use-Cases*. Jan. de 2020. url: <https://www.workato.com/the-connector/workflow-automation-2/>.
- [10] Zapier. *Zapier*. url: <https://zapier.com/>.
- [11] Zapier. *Learn key concepts in Zapier*. Jan. de 2021. url: <https://zapier.com/help/create/basics/learn-key-concepts-in-zapier>.
- [12] *Power Automate*. url: <https://emea.flow.microsoft.com/pt-pt/>.
- [13] Integromat. *Integromat*. url: <https://www.integromat.com/>.
- [14] Integromat. *Get Slack messages for new inbound emails, save email attachments to Dropbox and email text in Evernote*. url: <https://www.integromat.com/en/integration/37-get-slack-messages-for-new-inbound-emails-save-email-attachments-to-dropbox-and-email-text-in-evernote>.
- [15] Ifttt. *IFTTT*. url: <https://ifttt.com/>.
- [16] Automate.io. *Automate.io*. url: <https://automate.io/>.
- [17] Ashok Gudibandla e Author Ashok Gudibandla. *Twitter Lead Gathering*. Mai. de 2016. url: <https://automate.io/blog/automate-twitter-lead-generation/automate-twitter-leads/>.
- [18] *n8n.cloud*. url: <https://www.n8n.cloud/>.
- [19] *N8n.io*. url: <https://n8n.io/>.
- [20] *Workato*. url: <https://www.workato.com/>.
- [21] *Steps*. url: <https://docs.workato.com/recipes/steps.html>.
- [22] Bashar Nuseibeh e Steve Easterbrook. «Requirements Engineering: A Roadmap». Em: *Proceedings of the Conference on The Future of Software Engineering*. Limerick, Ireland: Association for Computing Machinery, 2000, pp. 35–46. isbn: 1581132530.

- [23] R.B. Grady e D.L. Caswell. *Software Metrics: Establishing a Company-wide Program*. Prentice-Hall, 1987. isbn: 9780138218447.
- [24] SAVE. url: <https://www.value-eng.org/>.
- [25] Kuniyoshi Urabe, John Child e Tadao Kagono. *Innovation and management: international comparisons*. W. de Gruyter, 1988.
- [26] Peter A. Koen, Heidi M. J. Bertels e Elko Kleinschmidt. «Managing the Front End of Innovation—Part I: Results From a Three-Year Study». Em: *Research-Technology Management* 57.2 (2014), pp. 34–43. doi: 10.5437/08956308X5702145.
- [27] Peter Koen et al. «Providing Clarity and A Common Language to the “Fuzzy Front End”». Em: *Research-Technology Management* 44.2 (2001), pp. 46–55. doi: 10.1080/08956308.2001.11671418.
- [28] Peter A. Koen, Heidi M. J. Bertels e Elko J. Kleinschmidt. «Managing the Front End of Innovation—Part II: Results from a Three-Year Study». Em: *Research-Technology Management* 57.3 (2014), pp. 25–35. doi: 10.5437/08956308X5703199.
- [29] Georg Von Krogh, Kazuo Ichijo e Ikujiro Nonaka. *Enabling knowledge creation: how to unlock the mystery of tacit knowledge and release the power of innovation*. Oxford University Press, 2000.
- [30] P. Koen et al. «Fuzzy Front End : Effective Methods , Tools , and Techniques». Em: 2002.
- [31] Molly Sloan. *4 Key Secrets Behind Zapier's Extraordinary Growth*. Dez. de 2020. url: <https://www.drift.com/blog/how-zapier-grew/>.
- [32] Integromat. *2019: Year in Review*. url: <https://www.integromat.com/en/blog/gaining-momentum-integromats-2019-year-in-review#:~:text=Number%20of%20users,%20December%202019:%20205,000>.
- [33] Integromat. *2020: Year in review*. url: <https://www.integromat.com/en/blog/integromat-2020-in-review>.
- [34] Thomas L. Saaty. *The analytic hierarchy process : planning, priority setting, resource allocation*. New York; London: McGraw-Hill International Book Co., 1980. isbn: 0070543712 9780070543713.
- [35] Thomas L. Saaty. «Some Mathematical Concepts of the Analytic Hierarchy Process». Em: *Behaviormetrika* 18.29 (jan. de 1991), pp. 1–9. issn: 0385-7417, 1349-6964. doi: 10.2333/bhmk.18.29_1.
- [36] Lawrence D. Miles. *Techniques of value analysis and engineering*. Eleanor Miles Walker, 1989.
- [37] Philip Kotler. *Marketing management: the millenium edition*. Prentice-Hall, 2000.
- [38] Tony Woodall. «Conceptualising 'Value for the Customer': An Attributional, Structural and Dispositional Analysis». Em: *Academy of Marketing Science Review* 12 (jan. de 2003).
- [39] A. Parasuraman e Dhruv Grewal. *Special issue on "serving customers and consumers effectively in the twenty-first century: emerging issues and solutions"*. Sage Periodicals Press, 1999.
- [40] Sugoto Bagchi e Bill Tulsie. «E-business models: integrating learning from strategy development experiences and empirical research». Em: *20th Annual International Conference of the Strategic Management Society*. 2000, pp. 15–18.
- [41] Ajit Kambil, A Ginsberg e M Bloch. «Rethinking Value Propositions». Em: *New York, NYU Center for Research on Information Systems* (1997).
- [42] Alexander Osterwalder e Yves Pigneur. *Business model generation: a handbook for visionaries, game changers, and challengers*. John Wiley e Sons, 2010.

- [43] Philippe B Kruchten. «The 4+ 1 view model of architecture». Em: *IEEE software* 12.6 (1995), pp. 42–50.
- [44] Gunnar Brataas e Peter Hughes. «Exploring Architectural Scalability». Em: *Proceedings of the 4th International Workshop on Software and Performance*. New York, NY, USA: Association for Computing Machinery, 2004, pp. 125–129. isbn: 1581136730.
- [45] Rajkumar Buyya, Rodrigo N. Calheiros e Amir Vahid Dastjerdi. *Big data: principles and paradigms*. Elsevier/Morgan Kaufmann, 2016.

Apêndice A

Método AHP

A.1 Escala Fundamental de Saaty

Tabela A.1: Escala fundamental de Saaty

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

A.2 Valores de IR para matrizes quadradas de ordem n

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

Figura A.1: Tabela de valores IR para matrizes quadradas de ordem n.

A.3 Estabelecer Prioridades Relativas

Sendo:

- A: Recursos Tecnológicos;
- B: Facilidade de Integração;
- C: Tempo de Desenvolvimento;

Tabela A.2: Matriz normalizada dos critérios do segundo nível

	A	B	C
A	$\frac{2}{3}$	$\frac{2}{3}$	$\frac{1}{2}$
B	$\frac{3}{11}$	$\frac{2}{7}$	$\frac{1}{3}$
C	$\frac{2}{11}$	$\frac{1}{7}$	$\frac{1}{6}$

A.4 Avaliar Consistência das Prioridades Relativas

O vetor prioridades (x) corresponde à coluna referente às prioridades relativas da tabela Y de prioridades relativas dos critérios, ou seja:

$$x = (0.54; 0.30; 0.16)$$

Considera-se que:

$$Ax = \lambda_{\max}x$$

Uma vez que A corresponde à matriz de comparação dos critérios do segundo nível representada na Tabela Y:

$$\begin{aligned} \begin{bmatrix} 1 & 2 & 3 \\ 0.5 & 1 & 2 \\ 0.33 & 0.5 & 1 \end{bmatrix} \begin{bmatrix} 0.54 \\ 0.30 \\ 0.16 \end{bmatrix} &\cong \lambda_{\max} \begin{bmatrix} 0.54 \\ 0.30 \\ 0.16 \end{bmatrix} \Leftrightarrow \\ \Leftrightarrow \begin{bmatrix} 1.62 \\ 0.89 \\ 0.49 \end{bmatrix} &\cong \lambda_{\max} \begin{bmatrix} 0.54 \\ 0.30 \\ 0.16 \end{bmatrix} \\ \Leftrightarrow \lambda_{\max} &= \text{average} \left\{ \frac{1.62}{0.54}, \frac{0.89}{0.30}, \frac{0.49}{0.16} \right\} \simeq 3.01 \end{aligned}$$

Sendo n o número de critérios, o índice de consistência (IC) é dado por:

$$IC = \frac{\lambda_{\max} - n}{n - 1} = \frac{3.01 - 3}{3 - 1} \simeq 0.005$$

De acordo com a tabela de valores de IR, Figura A.1, o valor de IR para 3 critérios é 0.58. Considerando a fórmula da Razão de Consistência (RC):

$$RC = \frac{IC}{IR} = \frac{0.005}{0.58} \simeq 0.009 < 0.1$$

A.5 Construir Matriz da Comparação Paritária

Sendo:

- X: Implementação de uma plataforma no Goidini, reestruturando o projeto fair-code n8n.io e utilizando o sistema de gestão de utilizadores do Goidini;
- Y: Implementação de uma plataforma dentro dos BO's / Conta E-goi, reestruturando o projeto fair-code n8n.io;
- Z: Implementação de uma plataforma no Goidini, totalmente independente (plataforma, lógica e conectores);

Tabela A.3: Matriz normalizada de recursos tecnológicos do segundo nível

	X	Y	Z
X	$\frac{10}{17}$	$\frac{8}{13}$	$\frac{1}{2}$
Y	$\frac{5}{17}$	$\frac{4}{13}$	$\frac{2}{5}$
Z	$\frac{2}{9}$	$\frac{1}{13}$	$\frac{1}{10}$

Tabela A.4: Matriz normalizada de facilidade de integração do segundo nível

	X	Y	Z
X	$\frac{4}{7}$	$\frac{3}{5}$	$\frac{1}{2}$
Y	$\frac{2}{7}$	$\frac{3}{10}$	$\frac{3}{8}$
Z	$\frac{1}{7}$	$\frac{1}{10}$	$\frac{1}{8}$

Tabela A.5: Matriz normalizada de tempo de desenvolvimento do segundo nível

	X	Y	Z
X	$\frac{6}{13}$	$\frac{6}{13}$	$\frac{6}{13}$
Y	$\frac{6}{13}$	$\frac{6}{13}$	$\frac{6}{13}$
Z	$\frac{1}{13}$	$\frac{1}{13}$	$\frac{1}{13}$

Apêndice B

Código de implementação

B.1 Ficheiro GenericFunctions

```
1 ...
2 export async function egoiApiRequest(this: IHookFunctions | IExecuteFunctions | IExecuteSingleFunctions |
3 ILoadOptionsFunctions, method: string, endpoint: string, body: any = {}, qs: IDataObject = {},
4 headers?: object): Promise<any> { // tslint:disable-line=no-any
5
6     const credentials = this.getCredentials('egoiApi') as IDataObject;
7
8     const options: OptionsWithUrl = {
9         headers: {
10             'accept': 'application/json',
11             'Apikey': `${credentials.apiKey}`,
12         },
13         method,
14         qs,
15         body,
16         url: `https://api.egoiaapp.com${endpoint}`,
17         json: true,
18     };
19
20     if (Object.keys(body).length === 0) {
21         delete options.body;
22     }
23
24     try {
25
26         return await this.helpers.request!(options);
27
28     } catch (error) {
29         throw new NodeApiError(this.getNode(), error);
30     }
31 }
32 ...
```

Listing B.1: função EgoiApiRequest

B.2 Método execute

```

1 async execute(this: IExecuteFunctions): Promise<INodeExecutionData[]> {
2   let responseData;
3   const returnData: IDataObject[] = [];
4   const items = this.getInputData();
5   const length = items.length as unknown as number;
6   const operation = this.getNodeParameter('operation', 0) as string;
7   const resource = this.getNodeParameter('resource', 0) as string;
8   for (let i = 0; i < length; i++) {
9     try {
10      if (resource === 'contact') {
11        if (operation === 'create') {
12          const listId = this.getNodeParameter('list', i) as string;
13          const email = this.getNodeParameter('email', i) as string;
14          const resolveData = this.getNodeParameter('resolveData', i)
15            as boolean;
16          const aField = this.getNodeParameter('additionalFields', i)
17            as IDataObject;
18
19          const body: ICreateMemberBody = {
20            base: {
21              email,
22            },
23            extra: [],
24          };
25
26          if (additionalFields.birth_date) {
27            additionalFields.birth_date = moment(additionalFields.birth_date as string).format('YYYY-MM-DD');
28          }
29
30          if (additionalFields.extraFieldsUi) {
31            const extraFields = (additionalFields.extraFieldsUi as
32              IDataObject).extraFieldValues as IDataObject[];
33            if (extraFields) {
34              body.extra = extraFields as unknown as [];
35            }
36          }
37
38          Object.assign(body.base, additionalFields);
39
40          responseData = await egoiApiRequest.call(this,
41            'POST', ` /lists/${listId}/contacts`, body);
42
43          ...

```

Listing B.2: Parte do método execute

B.3 Ficheiro Interfaces

```

1 export interface ICreateMemberBody {
2   base: {
3     email?: string;
4     first_name?: string;
5     last_name?: string;
6     cellphone?: string;
7     birth_date?: string;
8     subscription_status?: string;
9   };
10  extra: [];
11 }

```

Listing B.3: Ficheiro Interfaces

B.4 Ficheiro Docker-compose

```

1 version: "3"
2
3 services:
4   traefik:
5     image: "traefik"
6     command:
7       - "--api=true"
8       - "--api.insecure=true"
9       - "--api.dashboard=true"
10      - "--providers.docker=true"
11      - "--providers.docker.exposedbydefault=false"
12      - "--entrypoints.websecure.address=:443"
13      - "--certificatesresolvers.mytlsschallenge.acme.tlschallenge=true"
14      - "--certificatesresolvers.mytlsschallenge.acme.email=${SSL_EMAIL}"
15
16   ports:
17     - "443:443"
18     - "80:80"
19
20   volumes:
21     - ${DATA_FOLDER}/letsencrypt:/letsencrypt
22     - /var/run/docker.sock:/var/run/docker.sock:ro
23
24   n8n:
25     image: n8nio/n8n
26     ports:
27       - "127.0.0.1:5678:5678"
28     labels:
29       - traefik.enable=true
30       - traefik.http.routers.n8n.rule=Host(`${DOMAIN_NAME}`)
31       - traefik.http.routers.n8n.tls=true
32       - traefik.http.routers.n8n.entrypoints=websecure
33       - "traefik.http.routers.n8n.rule=PathPrefix(`/${SUBFOLDER}/{regex:$${/*}.*`)"
34       - "traefik.http.middlewares.n8n-stripprefix.stripprefix.prefixes=${SUBFOLDER}"
35       - "traefik.http.middlewares.n8n-stripprefix"
36       - traefik.http.routers.n8n.tls.certresolver=mytlsschallenge
37       - traefik.http.middlewares.n8n.headers.SSLRedirect=true
38       - traefik.http.middlewares.n8n.headers.STSSeconds=315360000
39       - traefik.http.middlewares.n8n.headers.browserXSSFilter=true
40       - traefik.http.middlewares.n8n.headers.contentTypeNosniff=true
41       - traefik.http.middlewares.n8n.headers.forceSTSHheader=true
42       - traefik.http.middlewares.n8n.headers.SSLHost=${DOMAIN_NAME}
43       - traefik.http.middlewares.n8n.headers.STSIncludeSubdomains=true
44       - traefik.http.middlewares.n8n.headers.STSPreload=true
45
46   environment:
47     - DB_TABLE_PREFIX=conta1
48     - DB_MYSQLDB_DATABASE=goidini
49     - DB_MYSQLDB_HOST=goidini.e-goi.com
50     - DB_MYSQLDB_PASSWORD=goidinipassword
51     - DB_MYSQLDB_PORT=3306
52     - DB_MYSQLDB_USER=goidiniuser
53     - N8N_BASIC_AUTH_ACTIVE=true
54     - N8N_BASIC_AUTH_USER
55     - N8N_BASIC_AUTH_PASSWORD
56     - N8N_HOST=${DOMAIN_NAME}
57     - N8N_PORT=5678
58     - N8N_PROTOCOL=https
59     - NODE_ENV=production
60     - N8N_PATH
61     - WEBHOOK_URL=https://${DOMAIN_NAME}${N8N_PATH}
62
63   volumes:
64     - /var/run/docker.sock:/var/run/docker.sock
65     - ${DATA_FOLDER}/.n8n:/home/node/.n8n

```

Listing B.4: Exemplo de implementação do ficheiro docker-compose.yml

B.5 Ficheiro .env

```
1
2 # Folder where data should be saved
3 DATA_FOLDER=/root/n8n/
4
5 # The top level domain to serve from
6 DOMAIN_NAME=n8negoidomain.com
7
8 # The subfolder to serve from
9 SUBFOLDER=conta1
10 N8N_PATH=/conta1/
11
12 # DOMAIN_NAME and SUBDOMAIN combined decide where n8n will be reachable from
13 # above example would result in: https://n8negoidomain.com/n8n/
14
15 # The user name to use for authentication – IMPORTANT ALWAYS CHANGE!
16 N8N_BASIC_AUTH_USER=user
17
18 # The password to use for authentication – IMPORTANT ALWAYS CHANGE!
19 N8N_BASIC_AUTH_PASSWORD=password
20
21 # Optional timezone to set which gets used by Cron-Node by default
22 # If not set New York time will be used
23 GENERIC_TIMEZONE=Europe/Berlin
24
25 # The email address to use for the SSL certificate creation
26 SSL_EMAIL=conta1@e-goi.pt
27
28
```

Listing B.5: Exemplo do ficheiro .env

Apêndice C

Inquérito de Satisfação



Inquérito de Satisfação e Usabilidade

Este inquérito foi desenvolvido no âmbito da Unidade Curricular TMDEI durante a realização da Tese de mestrado do aluno Francisco Bobiano (1160593), e tem como principal objetivo obter feedback relativamente ao grau de satisfação e usabilidade da plataforma E-goi by n8n.

Agradeço a disponibilidade.

[Inicie sessão no Google](#) para guardar o seu progresso. [Saiba mais](#)

Guião

O inquérito está dividido em duas secções.

A primeira secção apresenta questões relativas à utilização geral da plataforma.
A segunda secção apresenta questões relativas à criação de workflows com o node E-goi.

[Seguinte](#) [Limpar formulário](#)

Este conteúdo não foi criado nem aprovado pela Google. [Denunciar abuso](#) - [Termos de Utilização](#) - [Política de privacidade](#)

Google Formulários

Figura C.1: Início do inquérito

Utilização geral da plataforma

Considera o acesso à plataforma prático e intuitivo? *

	1	2	3	4	5	
Muito insatisfeito	☐	☐	☐	☐	☐	Muito satisfeito

A plataforma permite criar um workflow de forma fácil? *

	1	2	3	4	5	
Muito insatisfeito	☐	☐	☐	☐	☐	Muito satisfeito

Considera rápida e intuitiva a adição de um node à área de criação de workflows? *

	1	2	3	4	5	
Muito insatisfeito	☐	☐	☐	☐	☐	Muito satisfeito

Figura C.2: Início da secção de utilização geral da plataforma

A plataforma permite responder às necessidades relativas aos workflows que pretende automatizar? *

1 2 3 4 5

Muito insatisfeito ☐ ☐ ☐ ☐ ☐ Muito satisfeito

De 0 a 5 indique o grau de confiança a utilizar a plataforma. *

1 2 3 4 5

Muito insatisfeito ☐ ☐ ☐ ☐ ☐ Muito satisfeito

Considera o tempo de resposta das operações tolerável? *

1 2 3 4 5

Muito insatisfeito ☐ ☐ ☐ ☐ ☐ Muito satisfeito

[Anterior](#) [Seguinte](#) [Limpar formulário](#)

Figura C.3: Fim da secção de utilização geral da plataforma

Criação de workflows com o node E-goi

Considera de fácil aprendizagem o processo de configuração do node? *

	1	2	3	4	5	
Muito insatisfeito	☐	☐	☐	☐	☐	Muito satisfeito

Considera o node útil relativamente aos dados que pode enviar para a sua conta E-goi? (ex: adicionar um contacto a uma lista da sua conta E-goi) *

	1	2	3	4	5	
Muito insatisfeito	☐	☐	☐	☐	☐	Muito satisfeito

A criação de expressões é de fácil aprendizagem? *

	1	2	3	4	5	
Muito insatisfeito	☐	☐	☐	☐	☐	Muito satisfeito

Considera os workflows com o node E-goi relevantes? *

	1	2	3	4	5	
Muito insatisfeito	☐	☐	☐	☐	☐	Muito satisfeito

De 0 a 5 indique o grau de confiança com a utilização do node E-goi nos workflows. *

	1	2	3	4	5	
Muito insatisfeito	☐	☐	☐	☐	☐	Muito satisfeito

Anterior

Submeter

Limpar formulário

Figura C.4: Secção de criação de workflows com o node E-goi

Apêndice D

Respostas do Inquérito de Satisfação

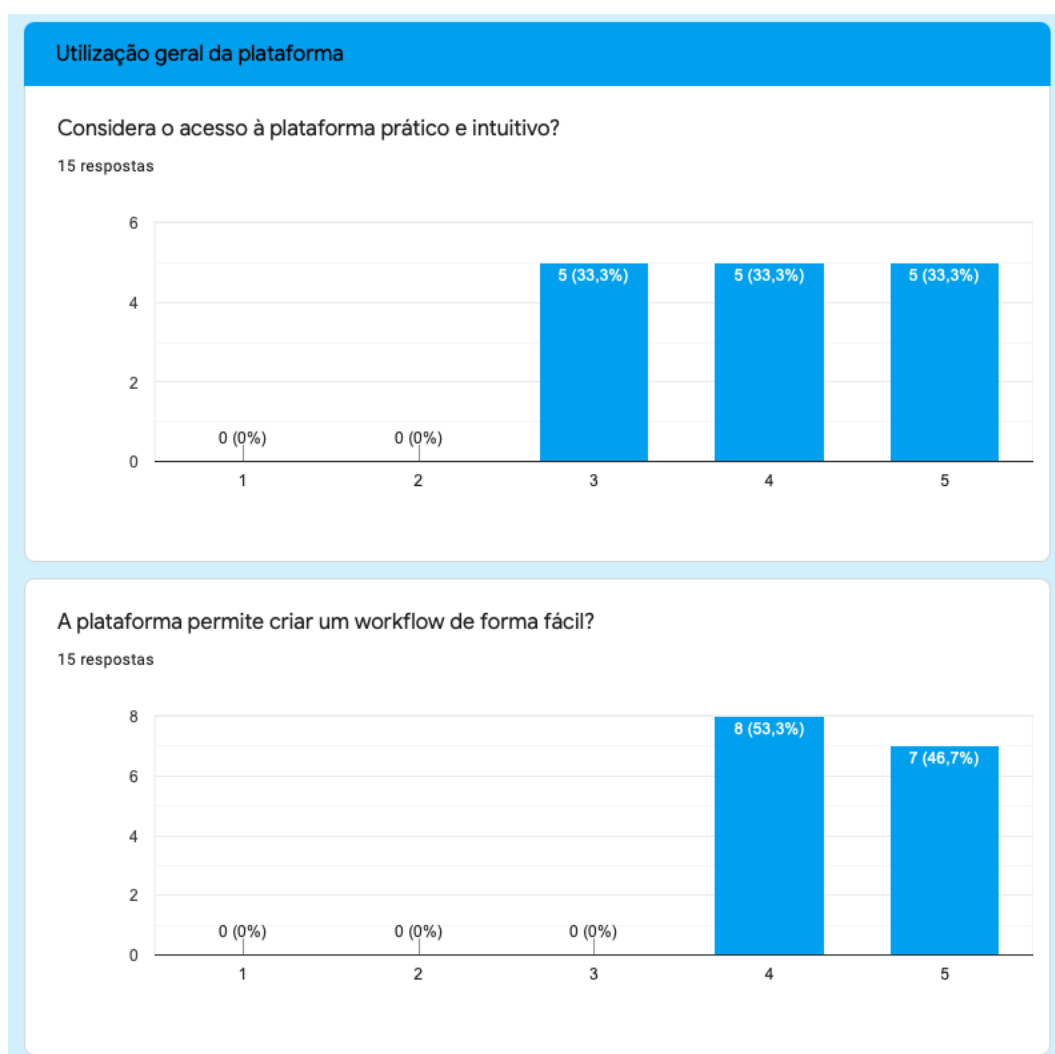


Figura D.1: Gráfico de barras das perguntas 1 e 2 da primeira secção

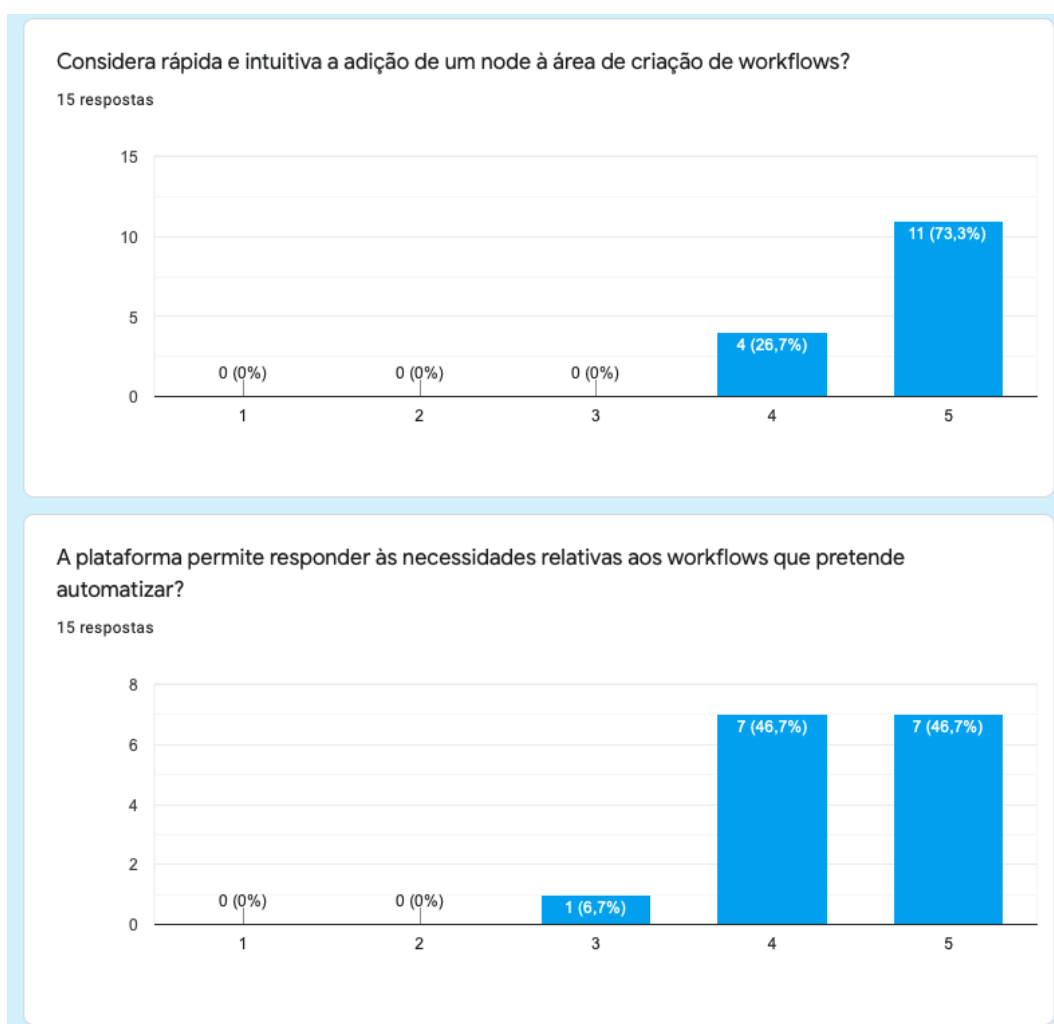


Figura D.2: Gráfico de barras das perguntas 3 e 4 da primeira secção

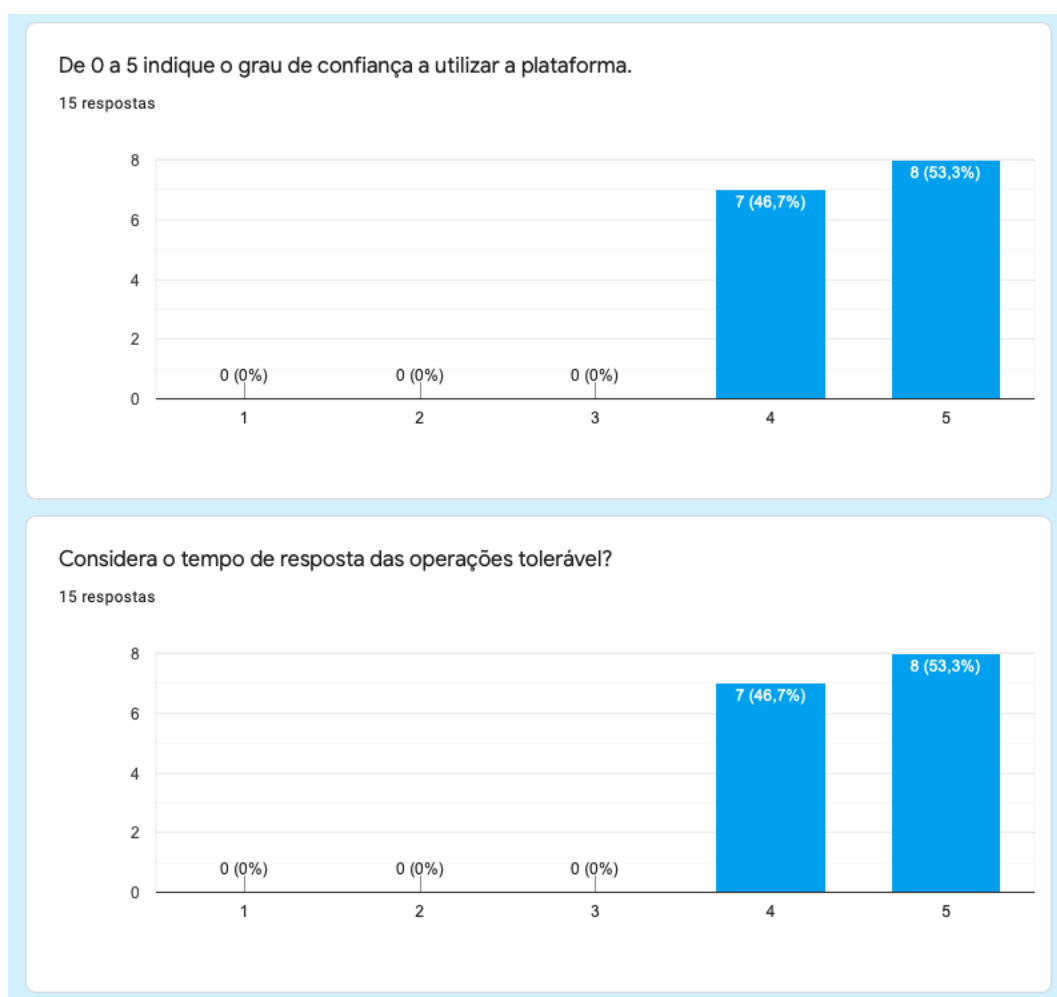


Figura D.3: Gráfico de barras das perguntas 5 e 6 da primeira secção

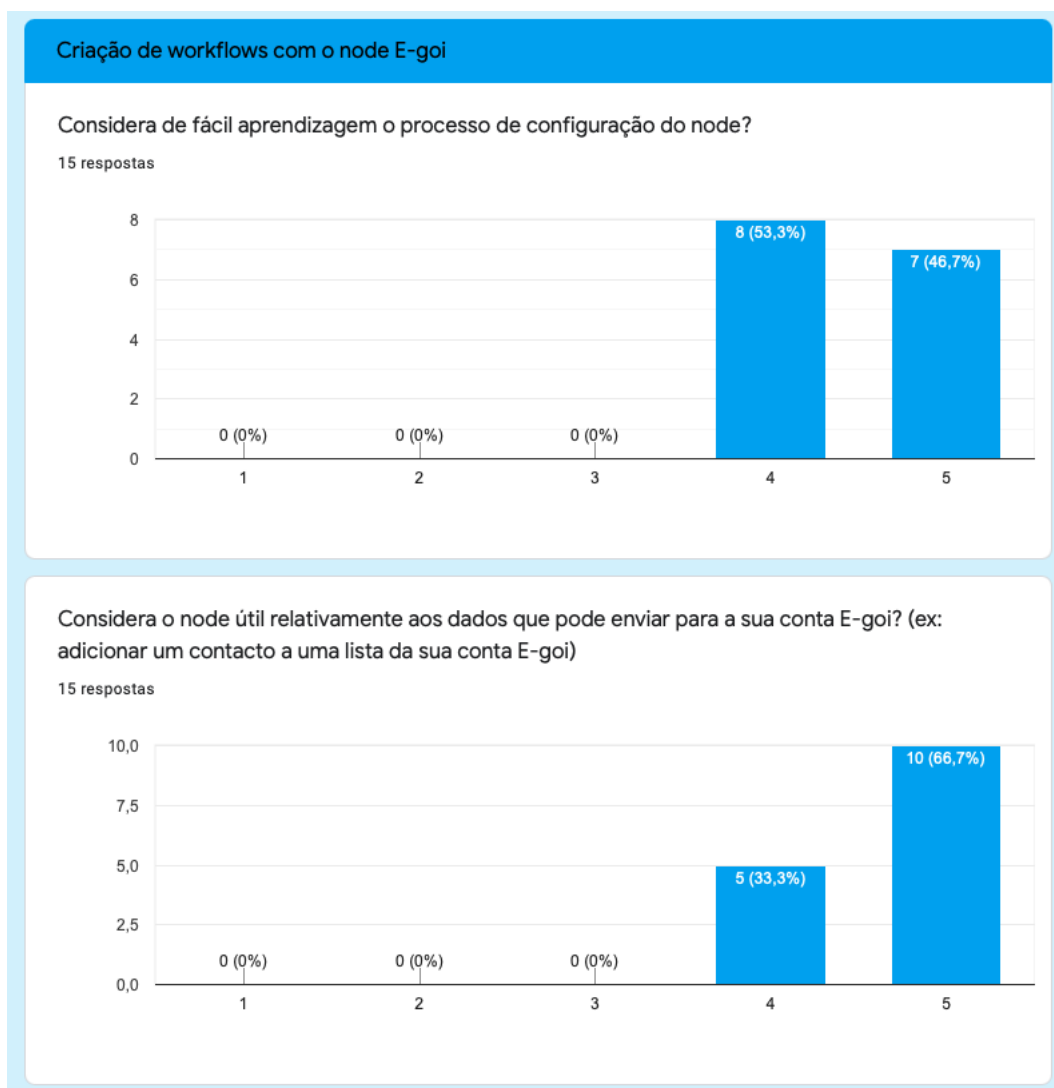


Figura D.4: Gráfico de barras das perguntas 1 e 2 da segunda secção

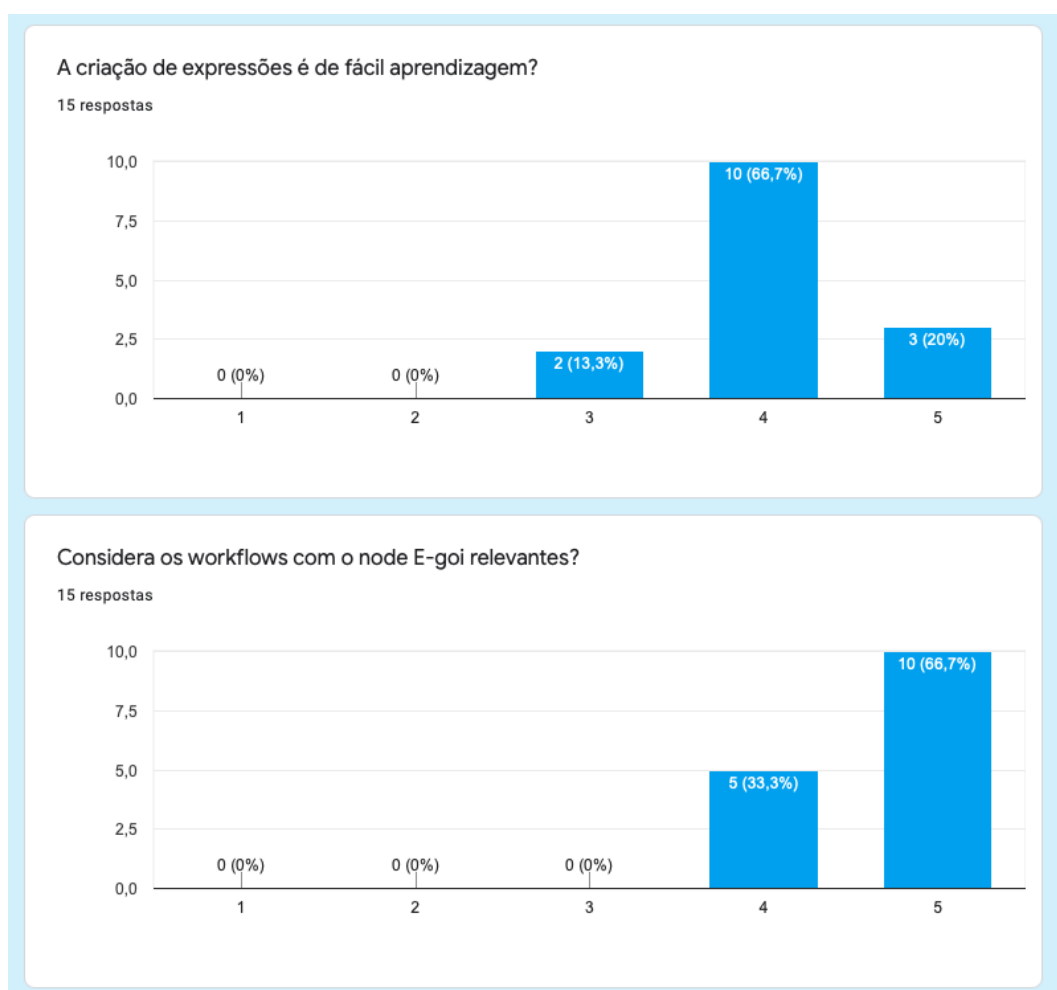


Figura D.5: Gráfico de barras das perguntas 3 e 4 da segunda secção

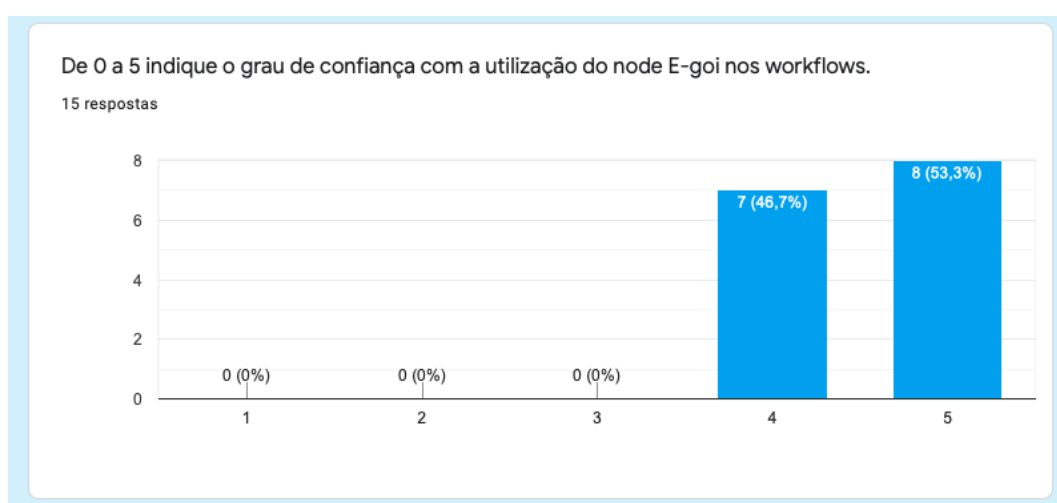


Figura D.6: Gráfico de barras da pergunta 5 da segunda secção