



Lisbon School
of Economics
& Management
Universidade de Lisboa

Mestrado em
MÉTODOS QUANTITATIVOS PARA A DECISÃO
ECONÓMICA E EMPRESARIAL

Documento Final

PROJETO

**DESENVOLVIMENTO DE UMA APLICAÇÃO LOW-CODE PARA GESTÃO
DE INTERVENÇÕES EM PROJETOS**

JOANA DANIELA SOUSA NÓBREGA

Orientação: Prof. Dr. Jesualdo Fernandes

Documento especialmente elaborado para a obtenção do grau de mestre

JULHO 2025

RESUMO

A transformação digital, impulsionada pela constante evolução dos Sistemas de Informação e pelo aparecimento de novas ferramentas de fácil adoção, como as ferramentas low-code, promove a eficiência e inovação dentro das organizações.

Este projeto foi realizado numa empresa do setor bancário, com vista ao desenvolvimento de um ecossistema integrado e funcional para gerir ações associadas a resolução de deficiências e/ou oportunidades de melhoria identificadas pelos diferentes órgãos de supervisão do sistema.

A empresa identificou como necessidades a falta de centralização, monitorização e visibilidade do estado da informação no âmbito do projeto. Para colmatar esta necessidade o objetivo deste projeto foi desenvolver uma aplicação em Microsoft Power Apps com o intuito dos vários interlocutores registarem as ações associadas a resolução de deficiências e/ou oportunidade de melhoria, *workflows* de alerta e de atualização automática dos dados dentro da aplicação recorrendo ao Microsoft Power Automate, e uma ferramenta de análise de dados utilizando o Microsoft Power BI. Todas estas ferramentas funcionam em conjunto permitindo a criação de uma solução para gerir projetos, agilizar e facilitar a análise das ações.

A metodologia utilizada para o desenvolvimento desta solução foi o Design Thinking composto por cinco fases: empatia, definição, idealização, protótipo e testes. Na etapa de protótipo são descritas todas as ferramentas utilizadas e o seu papel dentro do ecossistema desenvolvido.

Na fase dos testes, foram elaborados vários ajustes progressivos até a aplicação alcançar uma versão alinhada com as necessidades da organização. Embora atualmente esteja em produção, podem surgir novas necessidades de desenvolvimento que levarão a análise e implementação das mesmas na ferramenta atual. Desta forma, a aplicação mantém-se em constante evolução.

PALAVRAS-CHAVE: Automatização de Processos, Design Thinking, Eficiência Operacional, Low-Code, Transformação Digital

ABSTRACT

Digital transformation, driven by the constant evolution of information systems and the emergence of new, easy-to-adopt, tools such as low-code tools, promotes efficiency and innovation within organizations.

This project was carried out in a company in the banking sector, with a view to developing an integrated and functional ecosystem to manage actions associated with resolving deficiencies and/or opportunities for improvement identified by the different supervisory bodies of the system.

The company identified a lack of centralization, monitoring, and visibility of the status of information within the project as needs. To address the need, the objective of the project was to develop an application in Microsoft Power Apps so that the various stakeholders could record actions associated with resolving deficiencies and/or opportunities for improvement, alert workflows, and automatic data updates within the application using Microsoft Power Automate, and a data analysis tool using Microsoft Power BI. All these tools work together to create a solution for managing projects, streamlining and facilitating the analysis of actions.

The methodology used to develop this solution was Design Thinking, which consists of five phases: empathy, definition, ideation, prototyping, and testing. The prototyping phase describes all the tools used and their role within the developed ecosystem.

In the testing phase, several progressive adjustments were made until the application reached a version aligned with the organization's need. Although it is currently in production, new development needs may arise that will lead to their analysis and implementation in the current tool. In this way, the application remains in constant evolution.

KEYWORDS: Process Automation, Design Thinking, Operational Efficiency, Low-Code, Digital Transformation

GLOSSÁRIO

BPA – Business Process Automation

BPM – Business Process Management

DT – Design Thinking

ITR - Information Technology Request

LCAP - Low-Code Application Platform

RPA – Robotic Process Automation

ÍNDICE

Resumo	i
Abstract.....	ii
Glossário	iii
Índice	iv
Tabela de figuras	vi
Agradecimentos	vii
1. Introdução	1
2. Revisão da Literatura.....	3
2.1. Transformação Digital	3
2.2. Business Process Management e Business Process Automation	3
2.3. Robotic Process Automation	4
2.4. Plataformas Low-Code	5
2.5. Metodologia Design Thinking.....	8
3. Metodologia.....	10
3.1. Plano de Desenvolvimento	10
4. Descrição do Trabalho do Projeto	12
4.1. Introdução ao caso e objetivos.....	12
4.1.1. Empatia.....	12
4.1.2. Definir.....	14
4.1.3. Idealizar	14
4.1.4. Protótipo	16
4.1.4.1. Microsoft Lists.....	16
4.1.4.2. Figma.....	18
4.1.4.3. Microsoft Power Apps.....	20
4.1.4.4. Microsoft Power Automate	28
4.1.4.5. Microsoft Power BI	35

4.1.4.6. Microsoft SharePoint.....	37
4.1.5. Testes.....	38
5. Conclusões e Limitações	40
Referências	43

TABELA DE FIGURAS

Figura 1 - Ligações entre as ferramentas	16
Figura 2 - Interface ferramenta Figma.....	20
Figura 3 - Ambiente de desenvolvendo do Microsoft Power Apps	21
Figura 4 - Ecrã "Home" da aplicação	22
Figura 5 - Ecrã "Menu" da aplicação.....	22
Figura 6 - Ecrã "Consultar Ações" da aplicação	24
Figura 7 - Ecrã "Editar Ação" da aplicação.....	25
Figura 8 - Ecrã "Novo Relatório/Iniciativa" da aplicação	26
Figura 9 - Ecrã "Nova Ação" da aplicação.....	27
Figura 10 - Ecrã "Novo Responsável" da aplicação	28
Figura 11 - Fluxo para atualização de registos na aplicação	32
Figura 12 - Fluxo gerador de alarmísticas em Power Automate	35
Figura 13 - Esquema relacional Power BI.....	36
Figura 14 - Dashboard Power BI.....	37
Figura 15 - Página em SharePoint	38

AGRADECIMENTOS

Agradeço, inicialmente ao banco pela oportunidade de me permitir desenvolver competências numa área que considero de grande relevância e interesse profissional.

Quero agradecer ao Prof.º Doutor Jesualdo Fernandes por toda a disponibilidade e ajuda no decorrer da elaboração deste relatório.

Aos meus pais pela motivação e apoio ao longo de todo o meu percurso académico.
Uma ressalva especial para o meu namorado pela ajuda incondicional que me deu.

1. INTRODUÇÃO

Este projeto foi desenvolvido como trabalho final do mestrado em Métodos Quantitativos para a Decisão Económica e Empresarial no Instituto Superior de Economia e Gestão. O projeto foi elaborado numa empresa do setor bancário, que consistiu em desenvolver um sistema integrado utilizando várias ferramentas da Microsoft. Inicialmente foi criado uma aplicação low-code com recurso ao Power Apps, tendo como objetivo centralizar e gerir todas as informações relacionadas com o projeto, facilitando o acompanhamento do progresso, colaboração entre equipas e tomada de decisão informadas e garantia de que os objetivos do projeto são cumpridos dentro dos prazos estipulados.

Para dar suporte a esta ferramenta, foram desenvolvidos dois *workflows* utilizando o Power Automate. O primeiro fluxo permite atualizar automaticamente os registos da aplicação e o segundo fluxo é responsável por notificar os intervenientes do projeto sempre que um prazo for ultrapassado ou está prestes a ser atingido.

De forma complementar, foi elaborado um dashboard com recurso ao Power BI, que integra os dados provenientes da aplicação. Esta ferramenta permite uma visualização interativa das informações do projeto e proporciona aos utilizadores a capacidade de monitorizar os dados de forma visual e dinâmica. A combinação destas ferramentas permite uma gestão centralizada e eficiente do projeto.

As plataformas de desenvolvimento de aplicações low-code são uma séria aposta por parte das organizações nos correntes dias (Bock & Frank, 2021).

As rápidas mudanças nas exigências do mercado exigem respostas ágeis e flexíveis por parte das empresas e organizações. Esta necessidade por maior rapidez da resposta nas organizações pode ser de extrema relevância para que consigam acompanhar o mercado e manter-se competitivas, e responder de forma célere as necessidades regulamentares. Por fruto desta aceleração do meio envolvente, cada vez mais companhias do setor tecnológico investem em investigação e desenvolvimento de novas soluções low-code com o intuito de responder à procura emergente por estas aplicações (Phalake et al., 2023; Sahay et al., 2020).

Segundo a Forrester o valor do mercado das plataformas low-code pode atingir os 50 mil milhões de dólares até 2028 (Bratincevic et al., 2024). A Gartner estima que até 2025, 70% das novas aplicações desenvolvidas por parte das organizações serão criadas com recurso a tecnologias low-code (Cameron, 2022).

2. REVISÃO DA LITERATURA

2.1. Transformação Digital

Face à permanente aceleração e evolução dos Sistemas de Informação, a necessidade por encontrar novas abordagens que permitam maior agilidade às organizações gerou uma procura emergente pela transformação digital. Este tipo de transformação demonstra-se na otimização de modelos de negócio, processos, softwares e sistemas (Ebert et al., 2018; Schwertner, 2017). As novas ferramentas introduzidas pela transformação digital são elementos que permitem a reestruturação, otimização e automatização dos processos. Esta procura é particularmente relevante nas organizações cuja flexibilidade seja principal fator competitivo no mercado e por consequência ponha em causa a sua sobrevivência (Kraus et al., 2021).

As organizações estão atualmente inseridas em contextos de grande volatilidade, exigindo um maior aproveitamento da Inteligência Artificial e da automação da gestão dos processos empresariais, de modo a aumentar a eficiência e diferenciarem-se por meio de abordagens técnicas aplicadas com recurso a tecnologias avançadas (Romão et al., 2019).

O uso crescente das tecnologias inovadoras no mercado tem impulsionado a transformação digital, sendo que o Business Process Management (BPM) e Robotic Process Automation (RPA) são dois componentes integrantes desta transformação na medida em que a combinação sinérgica das mesmas, em determinados cenários, poderá revelar-se benéfica e incrementar o potencial das empresas (Kholiya et al., 2021).

2.2. Business Process Management e Business Process Automation

Segundo Weske (2019), o BPM (Business Process Management) compreende conceitos, métodos e técnicas que auxiliam o design, administração, configuração, execução e análise dos processos de negócio (Weske, 2019).

O BPM é uma atividade estratégica sobre os processos de negócio de uma organização no qual se pretende melhorar a eficiência desta, quer pelo aumento de produtividade quer pela redução de custos da organização. O BPM envolve um conjunto de etapas sequenciais que garantem que os processos estão bem estruturados e alinhados com as necessidades das organizações (Reinkemeyer Editor, 2020).

Inicialmente é realizada a etapa de modelação, onde são realizados os mapeamentos dos processos que existem e onde ocorre a definição das possíveis melhorias de processo a implementar. Na segunda etapa ocorre a execução, onde são implementados na prática os processos automatizados ou manuais que irão permitir a melhoria de processo. Na terceira etapa ocorre a monitorização, sendo nesta realizada o acompanhamento do desempenho dos processos. Na última etapa ocorre a otimização, onde os processos são ajustados de forma a aumentar a sua eficiência e evitar ou reduzir pontos de gargalo (Kunduru, 2023).

O Business Process Automation (BPA) é definido como a automação de processos e funções empresariais complexas, indo além da manipulação convencional dos dados e do registo de informações, geralmente por meio de uso de tecnologias avançadas. O foco do BPA está em operacionalizar o negócio, melhorando e otimizando os processos principais das organizações como a gestão de operações e serviços (Gartner, sem data; Melchert et al., 2004).

O BPA é uma área consolidada no BPM, estando relacionada com tecnologias e ferramentas como o Robotic Process Automation (RPA), Cognitive-Robotic Process Automation (C-RPA), Workflow Management Systems (WfMS), Enterprise Resource Planning (ERP) e o Blockchain (Moreira et al., 2023).

2.3. Robotic Process Automation

Uma das características da transformação digital é a automação em larga escala dos processos de negócio, em particular com o Robotic Process Automation (RPA), o uso deste tipo de software está atualmente em forte crescimento (Willcocks & Lacity, 2016).

O RPA é uma parte integrante do BPA que se concentra na automação de tarefas repetitivas e baseadas em regras, normalmente executadas por humanos (Sobczak, 2020). As tarefas realizadas podem incluir o tratamento ou inserção de informação, realizar cálculos, gerar relatórios ou integrar informações de múltiplas plataformas. Estas ferramentas de RPA geralmente não são aplicadas para automatizar processos de ponta a ponta, mas sim para subprocessos específicos ou até mesmo para elementos individuais dentro de um processo. O RPA são utilizados principalmente em processos operacionais e de suporte aos negócios, e menos implementados em áreas de atendimento ao cliente, sendo neste tipo de processo os Chatbots e Voicebots os mais utilizados (Sobczak, 2020).

Um dos grandes benefícios deste tipo de tecnologia passa pelo facto de libertar os colaboradores das organizações de tarefas monótonas e de pouco valor acrescentado, ao mesmo tempo que garante que os outputs são completos, corretos e consistentes entre tarefas e ajuda a garantir que várias tarefas sejam completadas mais rapidamente pois podem ser executadas sem necessidade de intervenção humana (IBM, sem data; Kholiya et al., 2021). Outros tipos de ferramentas de automação de processos, tal como as APIs, possuem esse tipo de interação. Um dos requisitos para o RPA é a necessidade das fontes de dados e o seu destino ter que ser altamente estruturado (Radke et al., 2020).

Tal como todas as ferramentas, o RPA apresenta vários desafios. A perda de conhecimento sobre um processo automatizado por robôs representa um desafio nas empresas, pois os colaboradores deixam de executar o processo manualmente e com o passar do tempo, perdem a familiaridade com os detalhes operacionais do processo automatizado. Como resultado, há uma dependência excessiva da tecnologia, o que dificulta a organização em diagnosticar processos ou realizar ajustes caso a automação falhe ou haja necessidade de ser alterada. Tendo em conta o contexto dinâmico e as mudanças frequentes em que as empresas estão inseridas, este desafio pode ser considerado impactante (Sobczak, 2020).

Outro desafio está no risco relacionado à segurança de informação, como este tipo de ferramentas executam tarefas entre diferentes sistemas que contêm dados confidenciais das empresas, podem estar mais vulneráveis a falhas, acessos não autorizados ou ataques externos (Sobczak, 2020).

A falta de algoritmos robustos de reconhecimento de erros que consigam verificar cada etapa do processo de forma continua também é considerada outra limitação deste tipo de ferramentas.

2.4. Plataformas Low-Code

Low-Code Application Platform (LCAP) é uma plataforma de aplicações que permite um desenvolvimento rápido de aplicações, a sua implementação, execução e gestão, utilizando abstrações de programação declarativas e de alto nível, como linguagens de programação baseadas em modelos e metadados, bem como implementações num único passo (Vincent et al., 2019). O termo low-code foi utilizado

pela primeira vez no contexto de transformação digital, pela Forrester Research, em 2014 (Richardson et al., 2014).

As LCAP do ponto de vista da arquitetura, consistem em quatro camadas principais: Camada de Aplicação, Camada de Integração de Serviços, Camada de Integração de Dados e Camada de Implementação. A Camada de Aplicação é composta pelo ambiente visual em que os utilizadores interagem diretamente, incluindo elementos como ferramentas e widgets destinados à construção da interface do utilizador. Nesta camada, o utilizador tem a capacidade de controlar o comportamento da aplicação, como a obtenção de dados de fontes externas e a sua manipulação para utilização na funcionalidade da plataforma ou em serviços externos, bem como a agregação de dados tendo por base regras definidas e a sua análise. A Camada de Integração de Serviços permite a ligação a diferentes serviços, com recurso a APIs e mecanismos de autenticação correspondentes. A Camada de Integração de Dados possibilita a manipulação dos dados de formas homogênea. Por fim, a Camada de Implementação é responsável pela gestão da contentização e orquestração das aplicações e funcionalidades de integração e implementação contínua (Sahay et al., 2020).

As plataformas low-code caracterizam-se por conjunto de tipos de objetos pré-criados que podem ser utilizados como recursos de rápida criação. São características comuns destas plataformas as interfaces visuais, tais como fluxos de trabalho ou outro tipo de componentes visuais, a inclusão de recursos que permitem a automação de processos de gestão, a sua facilidade de conectar com múltiplas aplicações integradas, bases de dados e serviços externos. Também é comum que estas ofereçam alguns tipos de controlo mais complexos para gerir a segurança e a conformidade da gestão da informação (Luo et al., 2021).

Estas soluções têm sido um veículo gerador da transformação digital das organizações. Segundo vários autores, estas permitem aumentos da produtividade dos colaboradores das organizações através da redução da necessidade de execuções manuais em controlo de processos, a redução dos seus custos por diminuição dos tempos necessários para construir aplicações e a diminuição de desenvolvedores especializados, uma maior facilidade na manutenção dos serviços permitindo que estas consigam

rapidamente se adaptar a novos contextos, sem a necessidade de código complexo (Phalake et al., 2023; Sanchis et al., 2020).

A redução da complexidade é outra vantagem relevante, como estas aplicações não são construídas do zero, tornam-se mais fáceis de desenvolver. Além disso, estas ferramentas facilitam o envolvimento de recursos humanos com perfis de negócio no desenvolvimento de aplicações, permitindo que pessoas sem conhecimentos técnicos participem ativamente no processo de construção de aplicações, facilitando a criação de soluções que respondem diretamente às necessidades e exigências do negócio, garantindo uma maior proximidade entre a tecnologia e o negócio. Este modelo de desenvolvimento promove também uma maior inovação, na medida em que os próprios utilizadores, que têm conhecimento aprofundado dos processos de negócio, possam criar ou personalizar solução técnicas sem necessidade de intermediários técnicos (Sanchis et al., 2020).

Ao utilizar uma infraestrutura clássica de desenvolvimento de software é necessário lidar com uma grande variedade de ferramentas separadas, como ferramentas de modelação, sistemas de gestão de bases de dados, frameworks de mapeamento objeto-relacional, editores de interfaces gráficas, assistentes de implementação e compilação, entre outras. As ferramentas low-code comparativamente às infraestruturas clássicas de desenvolvimento de software permitem incorporar todas estas ferramentas num único sistema, reduzindo o esforço para integrar os artefactos de implementação produzidos por diferentes tecnologias (Bock & Frank, 2021).

As principais desvantagens da utilização abusiva deste tipo de plataformas passa sobretudo pelas limitações de personalização, que podem levar à necessidade de código adicional para soluções que precisem de um maior nível de especificidade. Como estas plataformas são propriedade de organizações, pode tornar-se complexo qualquer tipo de migração para uma outra tecnologia. Um outro ponto bastante relevante é a escalabilidade, pois muitas plataformas podem não ser uma solução robusta quando o objetivo é criar uma aplicação com uma grande escala (Ihirwe et al., 2020; Luo et al., 2021).

Existem vários tipos de plataformas low-code, entre as principais destacam-se Microsoft PowerApps, OutSystems, Mendix e Oracle APEX. (Paul Vincent, 2019). O Microsoft PowerApps e o OutSystems caracterizam-se por ser soluções que não são open

source e que são de rápida aprendizagem para desenvolvimento, e suportam vários tipos de base de dados. O Mendix, embora seja uma plataforma open source, é necessária experiência em programação para a utilizar. Já o Oracle APEX também é uma ferramenta open source e se destaca pela sua facilidade de programação, no entanto a sua compatibilidade é restrita a base de dados Oracle (Ramos Alves & Gomes Soares Alcalá, 2022; Sahay et al., 2020; Talesra & G. S., 2021).

2.5. Metodologia Design Thinking

Recentemente, muitas organizações descobriram a eficácia da metodologia Design Thinking (DT) na criação de soluções. Esta abordagem combina a perspetiva dos utilizadores e a viabilidade tecnológica para criar e entregar soluções inovadoras que superem as expectativas tradicionais (Ambrose & Harris, 2010). O DT gera soluções potenciais para um problema ao proporcionar melhores produtos e serviços, além de um aumento na produtividade e melhorias operacionais (Wolniak, 2017).

O DT obteve grande reconhecimento e é atualmente considerado um paradigma inovador para resolver problemas em diversas áreas, como IT, Negócios, Educação e Medicina (Dorst, 2011).

Segundo vários autores, esta metodologia é composta por cinco etapas: Empatia, Definir, Idealizar, Protótipo e Teste, sendo que estas devem ser conduzidas em sequência, sem omitir nenhuma (Wolniak, 2017).

A primeira etapa intitulada “Empatia”, é utilizada para definir as características do público-alvo para o qual a solução será desenvolvida. Fornece um diagnóstico precoce das necessidades e expectativas dos potenciais utilizadores finais da solução e existe uma compreensão dos recursos técnicos que a empresa dispõe e do mercado envolvente.

Na segunda etapa, “Definir”, deverão ser especificadas as necessidades do utilizador final, realizando uma síntese das informações recolhidas na fase anterior para determinar e definir a extensão do problema. Durante esta fase, a abordagem deve ser mais orientada para a inovação e criatividade indo ao encontro das necessidades identificadas.

“Idealizar” consiste na terceira fase da metodologia, sendo que nesta etapa deverá ser selecionada e descrita a melhor solução.

A quarta etapa, “Protótipo” consiste na construção da solução escolhida na fase anterior, permitindo uma apresentação visual e obtenção de feedback rápido sobre o seu funcionamento. Esta etapa garante que a solução vai de encontro com as expectativas do utilizador final e minimiza o risco de potenciais falhas. No “Protótipo” a solução é testada e avaliada de forma a garantir a sua eficácia.

Na última fase, “Teste” a solução é apresentada aos utilizadores finais para obter opinião sobre a solução desenvolvida, sendo que os testes efetuados deverão ser feitos por um grande número de pessoas. Esta etapa consiste na verificação do desempenho da solução projetada em um ambiente real onde a solução será utilizada.

Gavin Ambrose e Paul Harris definem a metodologia DT em sete etapas, definir, pesquisa, idealizar, protótipo, selecionar, implementar e aprender (Ambrose & Harris, 2010). As etapas que são comuns entre os vários autores possuem a mesma definição.

Relativamente as etapas que são comuns entre os autores, a fase “Pesquisa” consiste na busca de informações relevantes que possam suportar o processo criativo, podendo assumir uma abordagem quantitativa, como a composição do público-alvo, ou qualitativa, com informação do perfil de cada grupo. A fase “Selecionar” corresponde ao momento em que uma das soluções propostas é selecionada para desenvolvimento, avaliando-se se a solução atende às necessidades e aos objetivos definidos, considerando também fatores relevantes como o custo e tempo. Na fase “Implementar” é entregue a solução final, e são prestados serviços de gestão de projeto com o objetivo de assegurar que os resultados estejam alinhados às expectativas, podendo ser necessárias revisões relacionadas à funcionalidade e à aparência visual (Ambrose & Harris, 2010).

3. METODOLOGIA

3.1. Plano de Desenvolvimento

O presente projeto foi desenvolvido seguindo a metodologia Design Thinking (DT). Esta metodologia é caracterizada pelo recurso à criatividade de quem elabora a solução (Ambrose & Harris, 2010), sendo uma metodologia que está alinhada com a visão do banco que procura fomentar a inovação tecnológica dentro da organização.

A liberdade na definição e implementação da solução permitiu adotar este tipo de metodologia, proporcionando inovação tecnológica para dentro da organização apresentando algo que não existia no departamento. O DT é composto por cinco fases, empatia, definição, idealização, protótipo e teste (Wolniak, 2017).

Para o desenvolvimento desta solução, fiz parte da equipa da Project Management Office (PMO) como responsável pelo desenvolvimento da solução, em articulação com os clientes internos da solução, pertencentes a diferentes áreas da empresa.

A Project Management Office (PMO) era responsável por guiar as reuniões com os utilizadores finais, tendo como responsabilidades:

- Recolha de necessidades com os utilizadores finais;
- Acompanhar o progresso da criação do ecossistema;
- Garantir o acesso a todas as ferramentas e às informações necessárias para o desenvolvimento do projeto;
- Reuniões de acompanhamento semanais para verificar o andamento do desenvolvimento da solução;
- Desbloquear dúvidas sobre o enquadramento do tema;
- Validações da aplicação antes de mostrar aos utilizadores finais;
- Recolha de feedback.

Como responsável pelo desenvolvimento da solução, as minhas responsabilidades eram mais técnicas, nomeadamente:

- Perceber as necessidades dos utilizadores;
- Compreender os dados e como os mesmos se interligam entre si;
- Definir quais são as melhores ferramentas para elaborar a solução;
- Desenvolver a solução;

- Apresentar a solução;
- Ficar com o ownership da solução, caso seja necessário adicionar novas funcionalidades ou ações de manutenção;
- Implementar as melhorias e/ou alterações recolhidas nas reuniões de feedback

Relativamente às responsabilidades dos clientes, elas são:

- Enquadramento e contextualização do tema;
- Definir as necessidades;
- Testar a solução;
- Garantir feedback para melhorias futuras e ações de manutenção.

O desenvolvimento desta solução foi realizado tendo por base as cinco etapas, defendidas por vários autores, que constituem o DT, nomeadamente: empatia, definição, idealização, protótipo e testes. A relação entre as etapas e o projeto serão contextualizadas no Capítulo 4.

4. DESCRIÇÃO DO TRABALHO DO PROJETO

Este capítulo tem como objetivo descrever todo o trabalho desenvolvido no âmbito do projeto e descrever as ferramentas utilizadas tal como a sua função em cada fase.

Inicialmente será apresentada uma introdução ao tema e os seus objetivos. Dentro deste tópico serão detalhadas as cinco fases da metodologia seguida, desde a recolha das necessidades dos utilizadores finais, até a elaboração da solução e, posteriormente, os testes à mesma. No subcapítulo “Protótipo”, a solução será detalhada, apresentando todo o ecossistema desenvolvido, incluindo todas as ferramentas utilizadas e como interagem entre si.

4.1. Introdução ao caso e objetivos

A gestão eficiente de projetos é essencial para garantir uma execução adequada de processos e otimização de recursos (Melchert et al., 2004). Neste contexto, este projeto foi realizado numa empresa do setor financeiro, onde foi identificado um problema na coordenação e monitorização das etapas de um projeto específico. Durante a perceção do problema, entendeu-se a necessidade de encontrar uma solução que facilitasse o acompanhamento das várias etapas de um projeto para os vários intervenientes e para os órgãos superiores da organização.

Para resolver este problema, foi desenvolvido um ecossistema com o objetivo de otimizar o fluxo de informação e gerir o progresso do projeto de forma centralizada e visível, reduzindo o tempo dispensado em encontrar formas de gerir e transmitir a informação e ao mesmo tempo ter uma solução eficiente e abrangente, isto é, desde o momento em que o dado entra em sistema, até à sua análise.

4.1.1. Empatia

Neste projeto, através de uma reunião inicial com os clientes, foi contextualizado e enquadrado o tema, identificando as necessidades, recolhido as expectativas e caracterizado o perfil dos utilizadores da solução.

O sistema bancário está sujeito a inspeções realizadas por diferentes entidades reguladoras, como o Banco de Portugal, auditores internos e externos, que avaliam a qualidade da informação presente nos sistemas da instituição bancária. Durante essas inspeções são identificadas deficiências e/ou oportunidades de melhoria. Estas

deficiências são identificadas em relatórios e comunicados à organização via e-mail ou por carta, para intervenientes específicos.

Num relatório podem ser identificadas várias deficiências e/ou oportunidades de melhoria, sendo que numa deficiência e/ou oportunidade de melhoria podem ser definidas várias ações para colmatar a resolução dessa deficiência e/ou oportunidade de melhoria.

Após a identificação de uma deficiência são designadas áreas específicas dentro da organização para abordar a resolução do tema, identificando um conjunto de ações dirigidas a diferentes colaboradores.

A análise inicial permitiu identificar diversas necessidades críticas dos *stakeholders*, entre as quais se destacam:

- Falta de centralização de toda a informação;
- Dispersão da informação pelos vários intervenientes nos seus ambientes pessoais;
- Falta de monitorização sobre o projeto, havendo dificuldades em identificar o estado das deficiências e/ou oportunidades e ações;
- Ausência de visibilidade dos órgãos superiores da instituição sobre o ponto de situação das deficiências e/ou oportunidades e das suas ações integrantes;
- Falta de comunicação entre os intervenientes;
- Ausência de visibilidade sobre a distribuição de responsabilidades;
- Falta de supervisão sobre o estado dos prazos de entrega.

Seguidamente, questionamos sobre as expectativas do cliente quanto à elaboração de uma solução, o qual identificaram:

- Uma ferramenta que centralize toda a informação e que seja acessível a todos os intervenientes de forma a facilitar a pesquisa de informação disseminada por múltiplas ferramentas isoladas e geridas por cada interveniente;
- Facilidade de uso e adoção das ferramentas, sem exigir formações complexas para serem usadas pelos intervenientes;
- Monitorização e atualização automática da informação provenientes de Information Technology Request (ITR) que está sujeita a resolução pela área de IT. Para a criação e seguimento de um ITR é necessário aceder a uma ferramenta

específica, sendo que a solução poderia colmatar a necessidade de consultar a ferramenta para monitorizar o estado do pedido;

- Visibilidade do estado da informação, permitindo identificar quem é o responsável por cada ação e deficiência, bem como o que é necessário para a sua completar a resolução de uma ação e consequentemente de uma deficiência e/ou oportunidade de melhoria;
- Alertas automáticos à medida que se aproxima a data definida para resolução da ação e/ou quando essa data seja atingida sem que a ação esteja concluída;
- Ferramenta que facilite a interpretação dos dados e que forneça uma melhor visualização do progresso da ação.

Com base nas necessidades e expectativas identificadas, concluiu-se que os clientes são, na sua maioria, profissionais orientados ao negócio, com grande foco em resultados operacionais e estratégicos, mas que possuem um grau de know-how técnico reduzido em tecnologias de informação e que demonstram uma resistência inicial ao tempo dedicado a adoção de uma nova solução complexa.

4.1.2. Definir

Nesta fase foi definido o problema ao qual queremos colmatar com a solução final. Após a recolha das necessidades foi identificado a falta de centralização, monitorização e visibilidade do estado da informação no âmbito do projeto, tendo sido definido como âmbito deste projeto a criação de um ecossistema centralizado que possibilite a integração automática da informação proveniente dos intervenientes no projeto e que seja gerida de forma automática um sistema de alarmísticas evidenciando casos de proximidade dos prazos definidos, e uma solução que permita a análise de todo o estado do projeto e o seu progresso.

4.1.3. Idealizar

Após uma análise das várias ferramentas disponíveis na organização, foi anunciada a existência de uma colaboração com a empresa da Microsoft, permitindo desta forma aceder a um vasto conjunto de ferramentas desta empresa.

Foi escolhida como ferramenta para inserção, consulta e edição de informação o Microsoft Power Apps. Com recurso a esta ferramenta é possível criar aplicações low-code, desenhar e implementar soluções que não requeiram um *know-how* extensivo de

programação possuindo um ambiente *user-friendly* para o desenvolvimento de aplicações. A escolha está alinhada com as expectativas dos utilizadores, por permitir criar uma solução fácil de usar e acessível a todos os intervenientes.

A criação de uma aplicação permite um ambiente centralizado e colaborativo entre as várias partes integrantes do projeto. Este tipo de solução proporciona a introdução da informação idêntica para todas as partes e um overview generalizado relativamente à forma como a informação é tratada e gerida, evidenciando as relações entre as ações, deficiências e/ou oportunidades de melhoria identificadas e respetivos responsáveis.

Para a elaboração da aplicação idealizou-se a criação de sete visões, sendo estas: entrada, índice, consultar ações, editar ações, inserir ações, inserir iniciativas e inserir responsáveis.

Como repositório de dados para gerir toda a informação, utilizou-se a ferramenta Microsoft List. Esta ferramenta facilita uma integração fácil com as Power Platforms (Power Apps, Power Automate), facilitando a sincronização automática em tempo real e automatização dos processos. As listas são simples de utilizar como fonte de dados e são eficientes para projetos com volume de dados moderado (Microsoft 365, s.d.).

Depois de definir o conteúdo de cada página da aplicação, idealizou-se o design através da ferramenta Figma. Esta ferramenta permite uma interface de fácil e rápida utilização para criação de layouts, e possibilita a exportação dos designs para diversos formatos (Figma, s.d.), sendo de utilização gratuita.

Para gerar os alertas automáticos e atualização dos dados entre diferentes ferramentas, foi idealizado a criação dos mesmos utilizando o Microsoft Power Automate. Esta ferramenta permite criar *workflows*, garantindo que os outputs são corretos e consistentes entre ferramentas e que várias tarefas, de intervenção humana, sejam automatizadas e executadas de forma simples sem necessidade de alocação de tempo por colaboradores.

Seguidamente foi idealizado um dashboard em Power BI alimentado pelas listas criadas em Microsoft Lists que possibilitasse a análise pelos intervenientes. Este dashboard conta com 3 visões: entrada, consultar deficiência e detalhe ações.

Tendo em conta que todas as ferramentas escolhidas são da Microsoft, com a exceção do Figma, existe a possibilidade de integração e sincronização ágil entre as mesmas,

automatizando tarefas e garantindo que os dados estejam sempre atualizados à periodicidade definida, permitindo uma interpretação dos mesmos baseada nas informações mais recentes.

Na figura 1 ilustra a idealização do funcionamento das ferramentas que englobam o ecossistema. O Figma é utilizado para construir o layout do Microsoft Power Apps, não apresentando uma relação de dependência por ser apenas a parte mais estética da aplicação. Por sua vez, o Microsoft Lists constitui o repositório de dados do ecossistema, sendo responsável por armazenar toda a informação criada no Power Apps e consultada através do Power Automate e Power BI, estabelecendo uma relação de dependência e transferência de dados entre estas ferramentas. O Power Automate envia alertas, sempre que aplicável, utilizando o Outlook para a comunicação por e-mail.

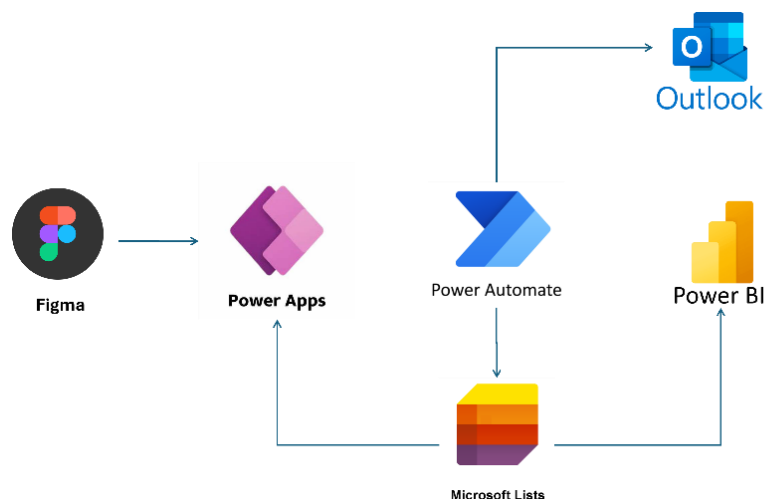


Figura 1 - Ligações entre as ferramentas

4.1.4. Protótipo

Após a seleção das ferramentas e a forma como estas interagem entre si, iniciou-se a fase de construção de todo o ecossistema e a implementação das funcionalidades planeadas. Nos próximos tópicos, irá ser detalhada cada ferramenta, explicando a sua utilidade e aplicação neste projeto.

4.1.4.1. Microsoft Lists

Nesta ferramenta foram desenvolvidas três listas. A primeira lista, intitulada “Iniciativas”, recolhe a informação das deficiências e/ou oportunidades de melhoria, onde

tem como campos: ID Relatório, Emissor, ID Deficiência, Descrição da Deficiência, Data de Resolução e Área Responsável. Seguem-se as definições de cada campo apresentado na lista:

- ID Relatório – Este campo representa o ID que é descrito no email/carta;
- Emissor – Representa o emissor das deficiências e/ou oportunidade de melhoria;
- ID Deficiência – Este campo é pedido ao utilizador para o inserir na aplicação com base na classificação da deficiência indicada no email/carta, que pode assumir formatos como “1A”, “1B”, ou, “1”, “2”, ou até outra sequência de números e letras provenientes do relatório. Para garantir que a chave gerada é única, é feito um tratamento a informação quando o utilizador insere a informação na aplicação, em que a chave final é composta pela combinação entre o ID Relatório e do ID Deficiência inserido na aplicação. Este campo representa o identificador único de cada registo, definindo a integridade referencial da lista.
- Descrição da Deficiência – Representa uma breve descrição de cada deficiência e/ou oportunidade de melhoria;
- Data de Resolução – Apresenta a data de resolução da deficiência e/ou oportunidade de melhoria;
- Área Responsável – Neste campo o responsável da deficiência e/ou oportunidade de melhoria é uma área em específico, e não uma pessoa singular.

A segunda lista, “Ações”, fornece informação das ações associadas a uma deficiência e/ou oportunidade de melhoria, apresentando como campos: ID Deficiência, ID Ação, Canal, Descrição da Ação, Ponto de Situação, ITR, Ask, Data Início, Responsável pela Ação, Status da Ação, Data Definida para Resolução e Data Efetiva de Resolução. Seguem-se o significado de cada campo:

- ID Deficiência: Representa o ID que é criado quando inserimos uma Iniciativa, não sendo passível de edição pelo utilizador;
- ID Ação: Representa o ID que é gerado sempre o utilizador cria uma ação, sendo composto pela combinação do ID Relatório e de um número que é

incremental. Este número é único e aumenta progressivamente a cada nova ação, garantindo que não há números repetidos entre novas ações e/ou deficiências e/ou oportunidade de melhoria;

- Canal: Representa o canal em que a ação está inserida;
- Descrição da Ação: Apresenta uma descrição sobre o conteúdo da ação;
- Ponto de Situação: Breve texto sobre o ponto de situação atual da ação;
- ITR: Número do ITR associado a resolução da ação, se aplicável;
- Ask: Número do Ask associado a resolução da ação, se aplicável;
- Data Início: Data de Início imputado ao começo da resolução da ação;
- Responsável pela Ação: Responsável singular pela resolução da ação;
- Status da Ação: Os vários estados em que pode estar a ação, sendo estes: Concluído, Em Curso, Por Iniciar;
- Data Definida para Resolução: Data em que o responsável imputa para conclusão da resolução da ação;
- Data Efetiva de Resolução: Corresponde a data em que o status da ação passa a “Concluído”.

A última lista, “Responsáveis”, contém campos de Nome do Responsável e Área. Esta lista apenas permite inserir novos responsável e imputar novas deficiências e/ou oportunidades de melhoria, ou ações.

As listas de Iniciativas e Ações, estão conectadas pela chave do ID Deficiência, que atua como ligação entre as duas listas, sendo que cada relatório tem várias deficiências, e cada deficiência pode estar associada a várias ações, possuindo uma relação de um-para-muitos.

Para a criação destas listas, foi pedido a recolha de informação existente com alguma estrutura de forma que fosse introduzida de uma só vez nas bases de dado que servem de fonte para todo o sistema. Esta informação necessitou de tratamento para ficar de acordo com a estrutura idealizada para a solução, sendo de seguida inserida no Microsoft Lists.

4.1.4.2. Figma

Após a definição do conteúdo de cada página da aplicação, foi elaborado o design através da ferramenta Figma, seguindo a estrutura de cores e os protocolos visuais exigidos pela instituição. Esta ferramenta permite uma interface de fácil e rápida

utilização para criação de layouts, e possibilita a exportação dos designs para diversos formatos (Figma, s.d.).

Com recurso à esta ferramenta foi desenvolvido o design das sete páginas da aplicação, incluindo os títulos das páginas, nomenclatura de cada objeto, botões de navegação entre as páginas e botões destinados à submissão de ações específicas.

Ao desenvolver o layout de cada página, foi possível escolher as medidas pretendidas para cada página, selecionar os objetos que ia utilizar, colocar os respetivos nomes dos campos e ajustar as cores. Para criar vários objetos, como por exemplo, um conjunto seguido de caixas que servem como filtros, é possível agrupá-los, permitindo assim que, caso seja necessário alterar de posição, consigamos realizar a alteração de todos simultaneamente.

Após a criação de todos os objetos e textos dentro das páginas, é necessário também agrupá-los, para no momento de fazer o “export” do layout, consigamos exportar todos os objetos que pertencem aquela página. Existe vários formatos de export, no entanto, optei pelo SVG por aparentar ter mais qualidade de imagem no momento do export.

No lado esquerdo da figura 2 é possível ver todos os objetos criados para a página atualmente selecionada, no lado direito conseguimos ver as medidas da página, respetivas cores e posição, e no centro inferior da página conseguimos criar os objetos, textos e ícones.

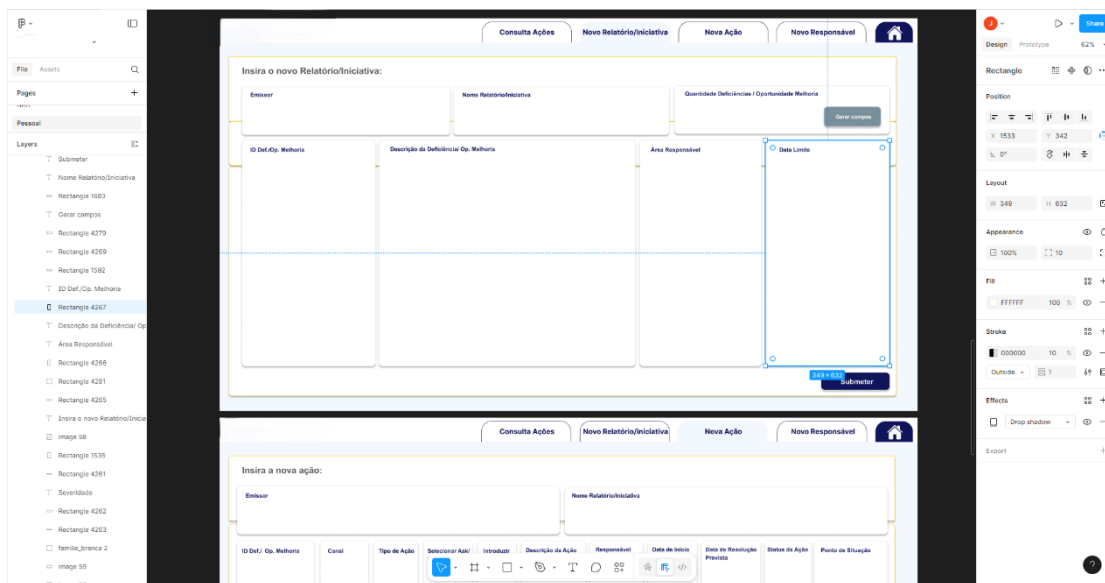


Figura 2 - Interface ferramenta Figma

Depois de criadas todos layouts das páginas e exportado, foi definido como “Background” em cada página do Microsoft Power Apps. Depois de definir o fundo de cada página, foram criados todos os objetivos, os quais foram sobrepostos ao design previamente elaborado, com as respectivas ações implementadas e funcionais.

À medida que houve o desenvolvimento destas páginas, através desta ferramenta é possível partilhar o layout com várias pessoas e/ou equipas, o que me permitiu ter feedback constante sobre o aspeto da aplicação por parte da PMO.

4.1.4.3. Microsoft Power Apps

Ao iniciar a criação da aplicação, o primeiro passo é escolher o tipo de aplicação: canvas app ou model-drive app. O canvas app é uma aplicação que não requer uma base de dados definida, permite definir manualmente o layout de cada página e possibilita a integração com vários sistemas. No model-driven apps existe a necessidade de uma base de dados em Microsoft Dataverse, sendo esta ferramenta uma base de dados com um conjunto de tabelas padronizadas, e as páginas são geradas automaticamente com base numa estrutura de dados. (Microsoft, s.d.).

Considerando todas as características do canvas app, a compatibilidade com o repositório de dados em Microsoft Lists, e considerando todo o ambiente que será construído, este tipo de aplicação destacou-se como a opção mais indicada.

Ao escolher a opção do canvas app, é exibido o ambiente de desenvolvimento da aplicação. No lado esquerdo estão os ecrãs disponíveis para navegação e edição. No lado superior esquerdo, encontra-se um conjunto de funcionalidades disponíveis que são controlados pelos scripts de código que são inseridos na barra do lado. No lado direito encontram-se as propriedades de cada item, permitindo ajustes como tamanho, cor, tipo de letra e ações específicas. Na figura 3 podemos ver o ambiente de desenvolvimento da aplicação.

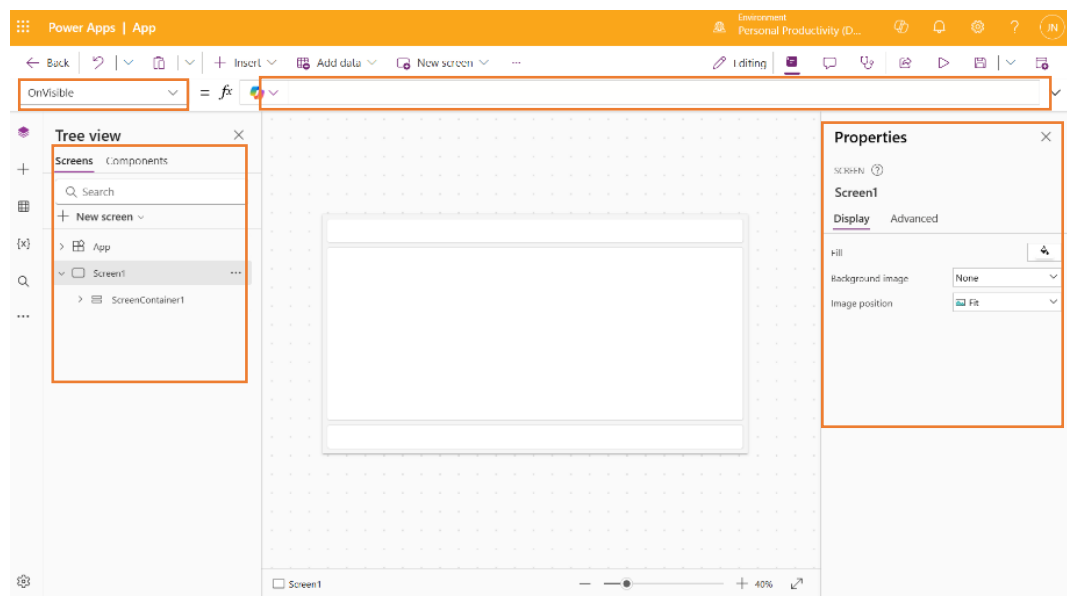


Figura 3 - Ambiente de desenvolvimendo do Microsoft Power Apps

Para a criação da aplicação foram utilizados os objetos pré-definidos pela ferramenta. No entanto, foi necessário implementar todos os scripts de código responsáveis pelas ações essenciais ao funcionamento da aplicação.

A primeira parte do desenvolvimento da aplicação foi a criação da página de entrada como mostrado na figura 4. Nesta página são apresentados dois objetos controlados por scripts de código que foram implementados: um deles identifica o utilizador que está a aceder a aplicação e personaliza a mensagem de “Bem-vindo”, tendo em conta o nome do utilizador; e o outro botão de “Entrar” redireciona para a página do índice.

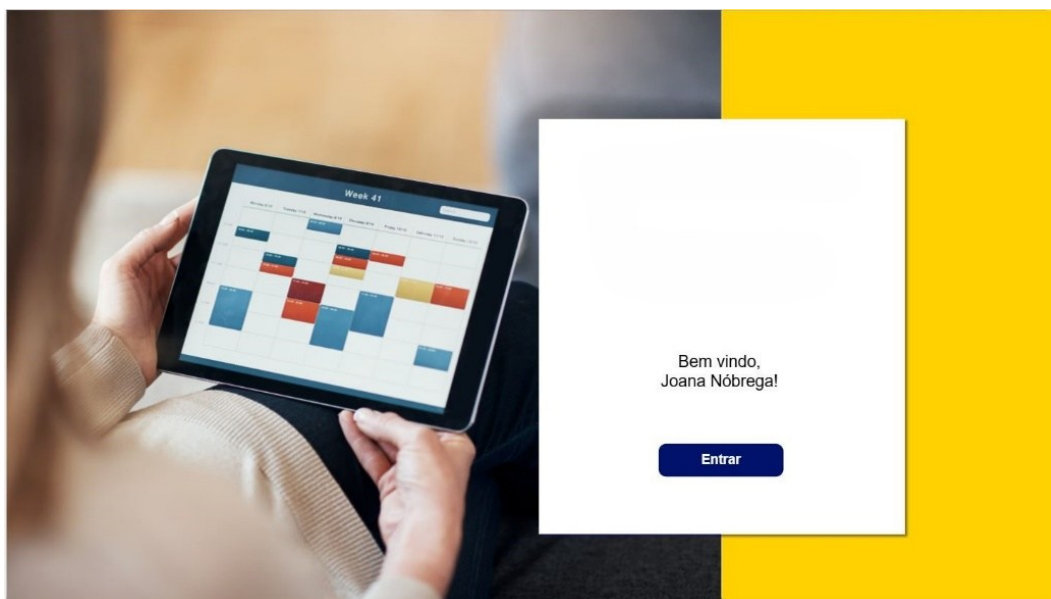


Figura 4 - Ecrã "Home" da aplicação

Seguidamente procedeu-se ao desenvolvimento da página do índice que permite ao utilizador escolher entre as várias páginas da aplicação, proporcionando uma visão abrangente e garantindo uma navegação intuitiva, como mostrado na figura 5. Os botões apresentados nesta view seguem a mesma lógica de redireccionamento para outras páginas.

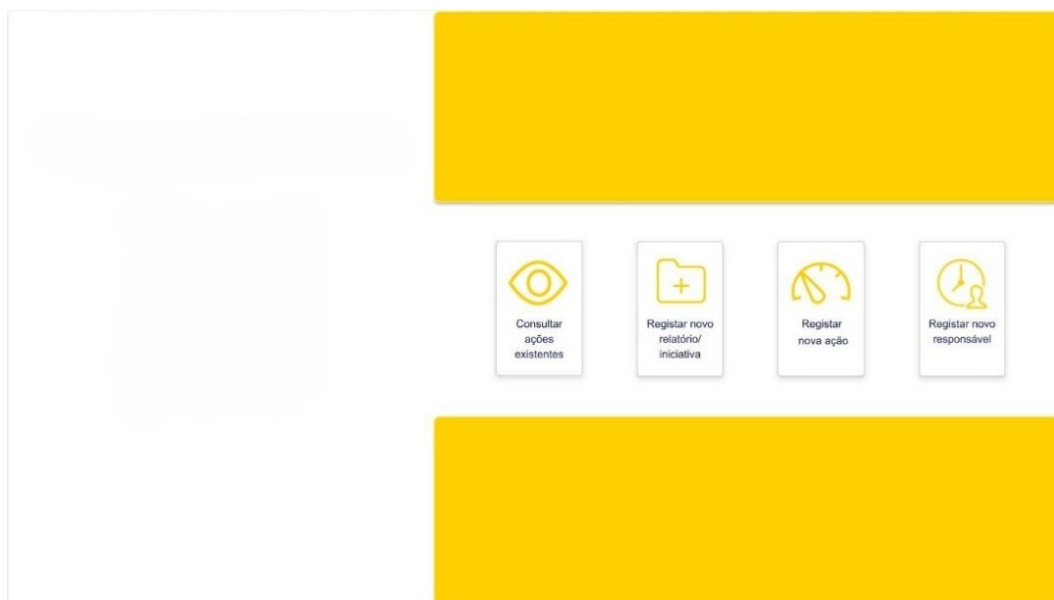


Figura 5 - Ecrã "Menu" da aplicação

Nas próximas páginas é apresentado uma barra localizada no canto superior de cada página, que permite ao utilizador navegar entre as páginas sem necessidade de

retornar ao menu principal. Além disso, facilita a identificação da página em que o utilizador se encontra, destacando-a com um fundo azul-claro.

No ecrã “Consultar Ações”, figura 6, são exibidas todas as ações e as respetivas deficiências e/ou oportunidades de melhoria com todas as suas características. A partir dos várias *dropdowns*, ou seja, do menu suspenso que apresenta uma lista de opções a escolher, é possível filtrar o conjunto de dados das ações apresentadas na tabela. Cada *dropdown* possui código que acede ao repositório de dados e retorna os valores distintos a serem exibidos de cada coluna. Entre os *dropdowns* existe lógica de filtragem cruzada, permitindo que os valores disponíveis sejam ajustados com base nas seleções anteriores. Os filtros também funcionam de forma independente. No canto superior direito é possível eliminar todos os filtros aplicados na tabela, selecionado o ícone “Reset”.

A tabela apresentada é uma combinação da lista das Iniciativas e das Ações. Para apresentar esta visualização, foram implementados vários códigos que acedem as características da respetiva deficiências e/ou oportunidade de melhoria, com base no ID Deficiência, que estabelece a ligação entre as duas listas. A filtragem da tabela é controlada pelos *dropdowns*, os quais possuem código que permite aplicar os filtros tanto de forma combinada como individual.

Para cada ação temos a opção “Editar” e “Eliminar” com um ícone, que ao carregar em “Editar” é redirecionado para a página das edições. Ao eliminar, é criado um pop-up com a ação que queremos eliminar e o utilizador confirma se tem a certeza que pretende eliminar a ação.

Figura 6 - Ecrã "Consultar Ações" da aplicação

Na figura 7, é apresentado o ecrã das “Edições”, onde permite ao utilizador editar uma ação específica, podendo apenas modificar os campos “Status da Ação”, “ITR”, “Ask” e “Ponto de Situação”, permanecendo todos os outros campos bloqueados de edição, garantindo um maior controlo da segurança da informação.

Após concluir as alterações, o utilizador seleciona o botão “Submeter” para efetuar as devidas alterações aos dados, sendo aplicado código para aceder a lista e fazer as devidas alterações nos dados. Contudo, se não quiser alterar os dados pode retroceder no botão “Consultar ações”, no canto superior da página.

Figura 7 - Ecrã "Editar Ação" da aplicação

Na figura 8, é exibido o ecrã “Novo Relatório/Iniciativa” sendo que o objetivo é o utilizador inserir as deficiências e/ou oportunidade de melhorias associadas a um relatório. Após preencher o emissor e o nome do relatório, escreve no campo “Quantidade de Deficiências e/ou Oportunidade de Melhoria” a quantidade de deficiências e seleciona o botão “Gerar campos”, que gera quantas linhas o utilizador decidir.

Após submeter a informação, o código implementado garante que, para cada deficiência e/ou oportunidade de melhoria é registado uma linha na lista, repetindo o Emissor e Nome do Relatório/Iniciativa para cada. Todos os campos apresentados na figura são todos de preenchimento obrigatório. Caso o utilizador deixe algum campo em branco, a submissão é impedida e é exibida a mensagem: “Necessita preencher todos os campos”. A informação apenas é submetida com sucesso após exibida a mensagem: “Novo Relatório/Iniciativa registada com sucesso!”.

Figura 8 - Ecrã "Novo Relatório/Iniciativa" da aplicação

No ecrã da figura 9, o objetivo é o utilizador inserir as ações realizadas para fazer face a uma deficiência e/ou oportunidade de melhoria. Este ecrã disponibiliza um conjunto de campos a preencher, e que após a submissão estará disponível para consulta no ecrã “Consulta Ações”.

Para inserir uma nova ação, o utilizador deve inicialmente selecionar os campos do emissor e do nome do Relatório/Iniciativa, sendo o campo “ID Def./ Op. Melhoria” filtrado posteriormente, tendo em conta as opções que existem com o emissor e relatório selecionado apresentando uma filtragem cruzada entre estes *dropdowns*.

Para inserir mais do que uma ação para a mesma deficiência e ou/oportunidade de melhoria, o utilizador deve carregar no botão inferior do lado esquerdo “Nova linha”, inserindo novas linhas quantas vezes carregar nesse botão. Para apagar linhas, basta selecionar o botão “Apagar linha”.

O *dropdown* “Tipo de Ação” tem como opções: “Ask”, “ITR”, “Outros”. Se o utilizador selecionar “Outros”, os campos “Selecionar Ask/ITR já existentes” e “Introduzir Ask/ITR” são bloqueados, visto não ser aplicáveis. Se selecionar “ITR” ou “Ask”, deve selecionar no campo “Selecionar Ask/ITR já existentes”, o respetivo Ask/ITR apresentando uma filtragem de todos os disponíveis, ou se não existir ainda, escreve o respetivo no campo “Introduzir Ask/ITR”.

Se no *dropdown* “Status da Ação” for “Por Iniciar”, o “Ponto de Situação” é preenchido automaticamente com “Por iniciar”, visto não ser aplicável.

Figura 9 - Ecrã "Nova Ação" da aplicação

Depois de o utilizador selecionar o botão “Submeter”, este objeto aplica código que verifica as seguintes condições:

- O campo “Data de Resolução Prevista” e “Data de Início de Prevista” são superiores a data do dia atual;
- O campo “Tipo de Ação” for “Ask” ou “ITR”, o código confirma qual dos campos, entre “Selecionar Ask/ITR já existente” ou “Introduzir Ask/ITR”, contém valor, considerando apenas aquele que estiver preenchido. Seguidamente, os dados são enviados para a coluna correspondente, “Ask” ou “ITR”, com base no valor indicado no campo “Tipo de Ação”;
- Verifica se todos os campos de todas as linhas estão preenchidos.

A submissão da informação só ocorre se todas estas condições forem cumpridas, caso contrário, são apresentadas mensagens de alerta indicando a falta de informação ou a incoerência entre as datas escolhidas. Se todos os requisitos forem cumpridos a informação é submetida e a mensagem “Nova ação registada com sucesso!” é exibida.

Por fim temos o ecrã “Novo Responsável” que permite ao utilizador inserir novos responsáveis e o departamento que pertencem, como é mostrado na figura 10. Estes campos também são de preenchimento obrigatório, exibindo alertas quando a informação não está completa.

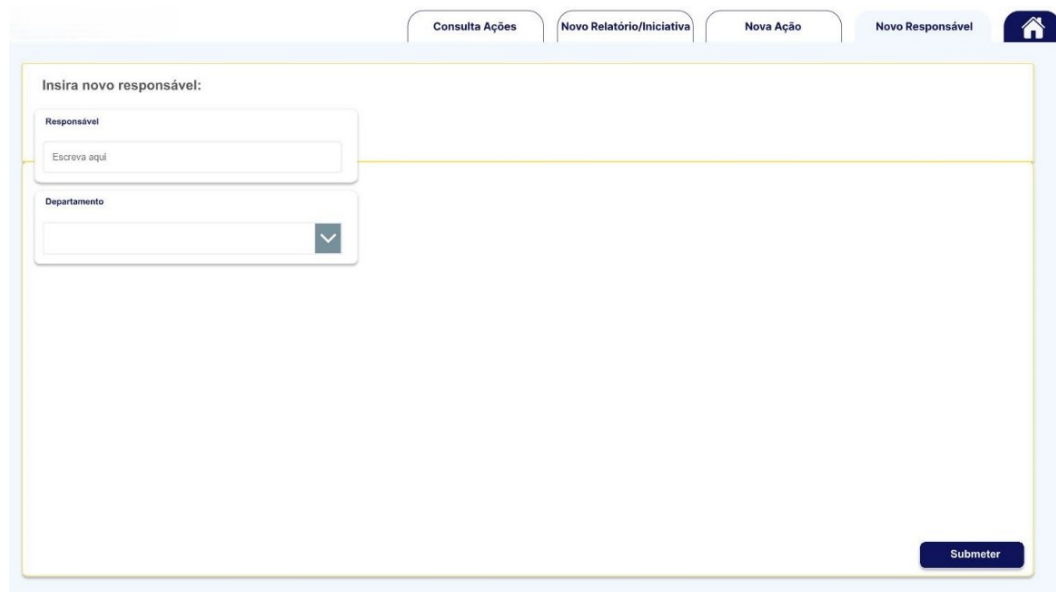


Figura 10 - Ecrã "Novo Responsável" da aplicação

Por uma questão de confidencialidade pelos dados da organização, todas as figuras encontram-se sem dados.

4.1.4.4. Microsoft Power Automate

Para iniciar a criação de um fluxo em Microsoft Power Automate, necessitamos de escolher que tipo de gatilho mais adequado ao fluxo. O gatilho constitui o evento em que desencadeia a execução do fluxo (Microsoft, 2025). Os principais tipos de gatilhos são:

- Agendado: Neste gatilho o objetivo é definir em que dia e hora queremos que o fluxo corra e podemos também definir a sua periodicidade de repetição, diária, semanal, mensal.
- Automático: Neste gatilho adicionamos uma ação que possa despoletar o fluxo, tal como: quando existe alguma modificação de um ficheiro em uma pasta, quando recebemos um email, quando existe nova informação, entre outros.

- Manual: Neste gatilho o fluxo é iniciado manualmente, sendo o utilizador a escolher quando pretende que o fluxo corra, selecionando um botão específico no momento pretendido.

A construção do primeiro fluxo serviu para colmatar uma das necessidades recolhidas dos clientes, o qual expectavam a centralização do estado de um Information Technology Request (ITR) sem ter de aceder a várias ferramentas para fazer o seguimento do mesmo.

Existem várias ações que são imputadas à resolução com base em um ITR. Um ITR é um pedido que as áreas podem fazer à área de IT, em que a resolução implica a intervenção dessa área, com atividades que só esta consegue desenvolver. Após a criação de um ITR, é imputado um número, em que a área regista esse número na aplicação em Power Apps no campo “ITR” associando este a uma ação.

Os pedidos a esta área são feitos através de uma ferramenta OnBoard, no entanto, não existe nenhum conector no Power Automate a esta ferramenta. Para colmatar esta limitação, foi solicitado à área de IT a possibilidade de disponibilização de uma extração do sistema num ficheiro de excel com informação referente ao estado dos vários pedidos feitos a esta área. Este ficheiro é disponibilizado numa pasta previamente acordada e atualizado diariamente pela área, fazendo overwrite do ficheiro anterior pela informação mais recente. Esta extração contém como campos:

- Work Type Item: Campo populado com “ITR” ou “Feature”, sendo que no primeiro só existe um registo, pois representa o estado atual do ITR, e o segundo representa os estados passados, ou seja, todas as mudanças de estados que ocorreram.
- Número do ITR: Este campo representa o número que é imputado ao ITR;
- Descrição do pedido: Representa a descrição do pedido;
- Estado: Os estados possíveis são: Avaliação, Cancelado, Certificação, Certificado, Fechado, Desenvolvimento, Execução, Planeamento, Produção e Reprovado;
- Data de criação: Data em que foi criado o pedido;
- Responsável: É registado quem fez o pedido;

Para iniciar a construção deste fluxo, conforme apresentado na figura 11, foi escolhido um gatilho automático, “When a new item is created or modified (properties only)”. Neste

gatilho verificamos se na pasta específica existe alguma modificação ou criação de um ficheiro e, se houver, o fluxo reconhece e começa a correr. Seguidamente temos a ação “Get file properties” para aceder ao conteúdo que está dentro do ficheiro que sofreu modificação ou criação.

Depois inicializamos uma variável escolhendo um nome e o tipo, que neste caso é string, para no fim do fluxo conseguirmos usar. Na “condition 1” é lido o nome do ficheiro modificado, e se for igual ao que queremos, então o fluxo passa o caminho “True”, caso contrário, se não cumprir a condição e entrar em “False”, o fluxo termina. Este passo garante caso seja inserido outro ficheiro que não seja o que pretendemos, o fluxo entra no ramo “False” e termina.

Se o ficheiro passar no “True”, a ação “List rows presente in a table” acede a informação presente no ficheiro de excel que foi criado e/ou modificado. Na ação seguinte “Filter array” é feito um filtro a informação presente no ficheiro de excel, filtrando apenas as linhas em que a coluna “Work Item Type” é igual a “ITR”, sendo que necessitamos de aceder apenas ao estado dos ITRs no momento atual.

A ação “Apply to each” tem como objetivo iterar sobre todos os outputs da ação anterior, “Filter array”. Esta ação filtra todas as linhas do ficheiro excel, e o fluxo percorre cada uma dessas linhas. Para cada linha filtrada, o fluxo executa os seguintes passos:

- Na ação “Get items” é feita uma pesquisa na lista “Ações” do Microsoft Lists, que serve como repositório de dados para a aplicação. Nesta ação, é verificado se existe algum registo na lista cujo campo “ITR” corresponda ao que está a ser atualmente comparado naquela iteração. Nesta ação o fluxo filtra a informação da lista procurando correspondência com o número do ITR em questão.
- Na seguinte ação “Apply to each” verificamos se a ação anterior devolveu algum registo. Se o número de registos encontrados for maior que zero, o fluxo segue para o ramo “True”, significando que encontrou pelo menos um registo com o número do ITR atual. Caso contrário, o fluxo segue para o caminho “False” e termina o fluxo para aquela iteração.

- Se entrar no ramo “True”, é aplicado a ação “Apply to each 2” que percorre todos os registos devolvidos pela ação “Get items”. Para cada um dos registos encontrados:
 - Na ação “Set variable” é atribuída uma variável previamente definida, que representa o estado ITR, convertendo-o para os estados utilizados na aplicação, sendo a lógica de transformação a seguinte:
 - Se o estado for Fechado, Reprovado, Certificado ou Cancelado, a variável assume o valor “Concluído”;
 - Se o estado for: Certificação, Desenvolvimento, Execução, Planeamento ou Produção, a variável assume o valor “Em curso”;
 - Se o estado for Avaliação, a variável assume “Por iniciar”;
 - Para outros estados não previstos no código, o valor original do mesmo é mantido.
 - Seguidamente, na ação “Update item”, é feito um update do estado dos ITRs na lista “Ações”. O campo “Status da Ação” recebe o valor da variável que foi convertida anteriormente. Sempre que o “Status da Ação” for “Concluído” o campo da “Data Efetiva de Resolução” é atualizado com a data do dia atual, caso contrário fica em branco.

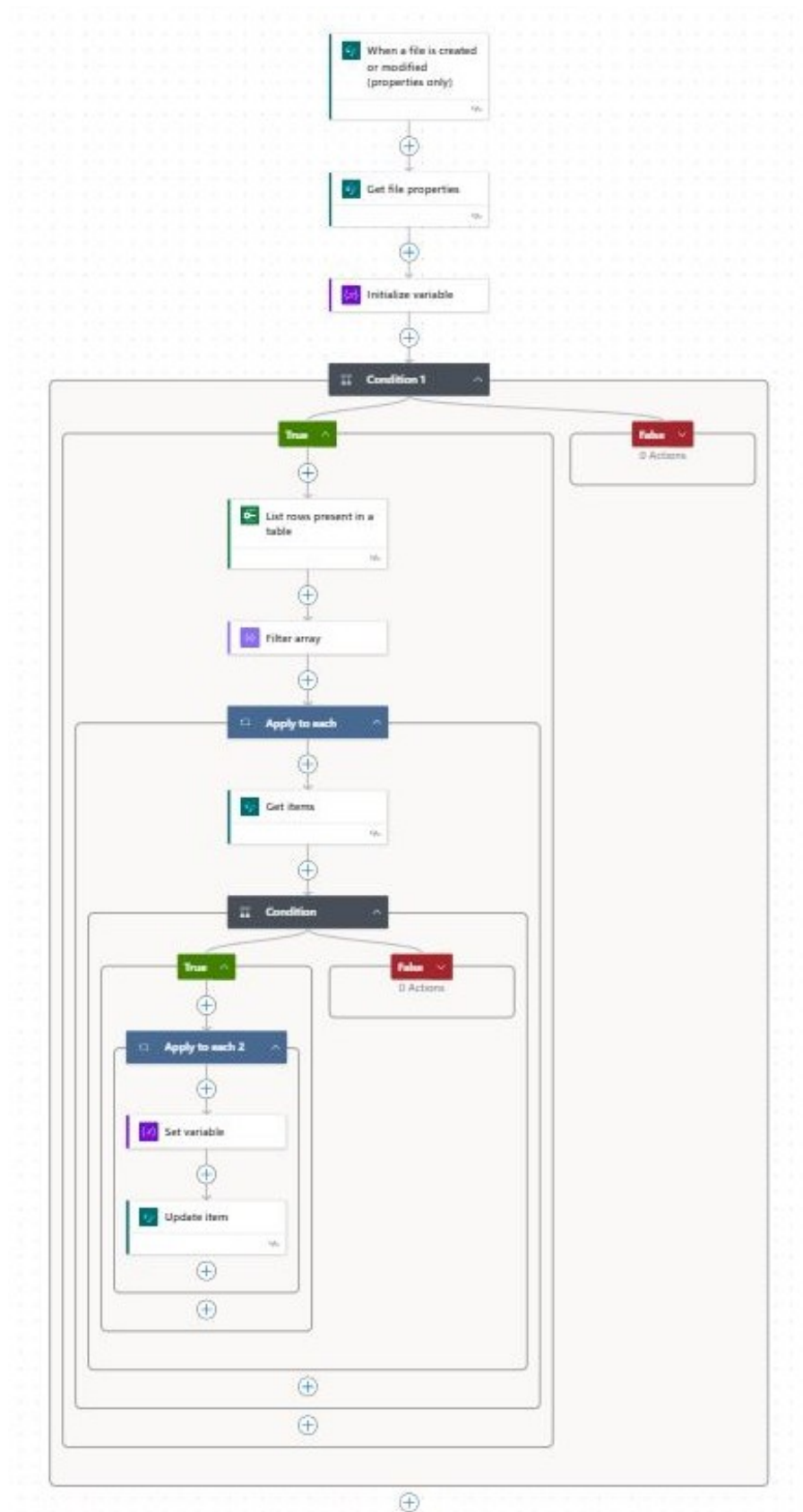


Figura 11 - Fluxo para atualização de registos na aplicação

O segundo fluxo desenvolvido atende à necessidade de notificar os diferentes intervenientes quando as ações registadas na aplicação estão prestes a atingir o prazo final, ou já ultrapassaram essa data e o status dela é diferente de concluído.

Para criar este fluxo, escolhemos um gatilho "agendado", por ser o mais pertinente neste tipo de fluxo, programando-o para ser executado todos os dias às 10 horas da manhã, realizando as seguintes instruções, conforme mostrado na figura 12:

- Na ação "Get Item Responsáveis" o fluxo acede a lista dos Responsáveis.
- A ação "Apply to each" tem como objetivo iterar sobre todos os Nomes dos Responsáveis presentes na lista dos Responsáveis, executando as ações subsequentes individualmente para cada responsável:
- Na ação "Get Item Ação" o fluxo acede a lista das Ações, e filtra todos os registos da lista em que o responsável é igual ao que está a ser iterado no momento, no campo "Filter Query" que é programado dentro da ação.
- Na ação "Filter array" é filtrado todos os registos da ação anterior em que:
 - O "Status da Ação" é diferente de concluído;
 - A data definida para resolução é inferior ao dia de hoje ou diferença entre a data definida para resolução e o dia de hoje é de 10 dias ou a diferença entre data definida para resolução e o dia de hoje é de 5 dias;
- Na ação "Condition" é verificado se o tamanho do output da ação "Filter array" é superior a zero. Caso afirmativo, indica que existem registos na lista, direcionando o fluxo para o ramo "TRUE" da condição, sendo executadas o seguinte conjunto de etapas:
 - Na ação "Select" é reorganizado os dados filtrados na ação "Filter Array", se aplicável. Nesta ação é selecionado apenas as informações relevantes de cada ação, como o Tema, ID Iniciativa, Emissor, Descrição da Ação, Data definida para Resolução e acrescenta um novo campo intitulado "Situação", que indica se a ação está:
 - 10 Dias para o Prazo Final: Se o "Status da Ação" for diferente de "Concluído" e a diferença entre a data atual e a data definida para resolução seja igual a 10.

- 5 dias para o Prazo Final: Se o "Status da Ação" for diferente de "Concluído" e a diferença entre a data atual e a data definida para resolução seja igual a 5.
 - Fora de Prazo: Se o "Status da Ação" for diferente de "Concluído" e data definida para resolução seja inferior a Data Atual.
- Na ação "Create HTML Table", o output da ação anterior é convertido em uma tabela em formato HTML, permitindo organizar as características das ações em uma tabela com os campos: Tema, ID Iniciativa, Emissor, Descrição da Ação, Status da Ação, Data definida para Resolução e Situação.
- Na ação "Send an email (V2)" é enviado para o Responsável atualmente iterado a respetiva tabela criada na ação anterior com as respetivas ações e detalhe criados na ação anterior.
- Caso a condição não entre no ramo "TRUE", é iterado o próximo responsável, e assim sucessivamente.

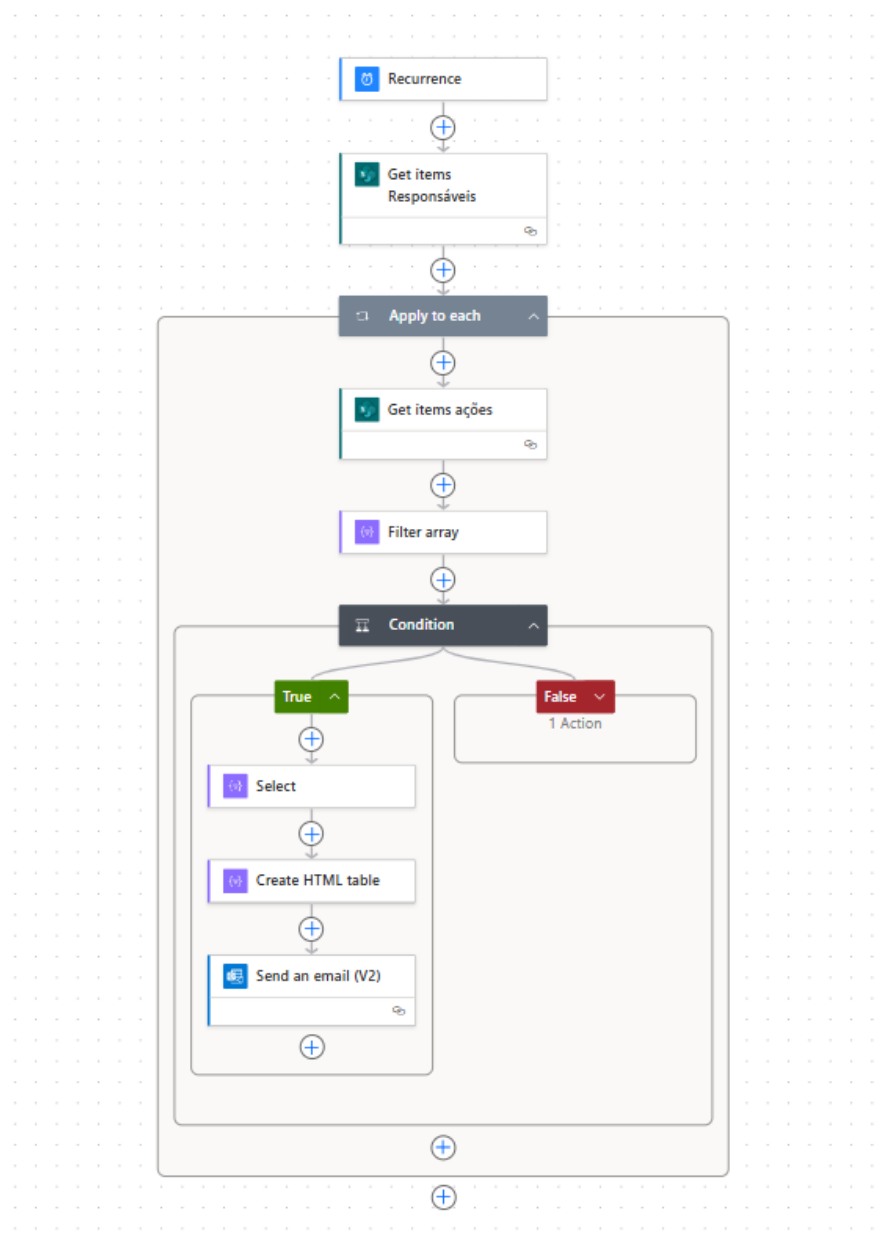


Figura 12 - Fluxo gerador de alarmísticas em Power Automate

4.1.4.5. Microsoft Power BI

Para o desenvolvimento desta solução, foi também como objetivo a criação de uma ferramenta visual para análise de dados, sendo para isso criado um dashboard em Power BI.

Com recurso a esta ferramenta é possível ver a distribuição dos dados visualmente, conseguindo identificar padrões e tendências e permitindo uma análise interativa com os dados.

Na figura 13 é apresentado o modelo de dados implementado. A tabela “Iniciativas” tem uma relação de um-para-muitos para a tabela “Ações”, ligando-se pelo ID Deficiência. A relação da tabela “Responsáveis” e “Ações” é de um-para-muitos pois um responsável pode estar alocado a muitas ações, ligando-se pelo Nome do Responsável.

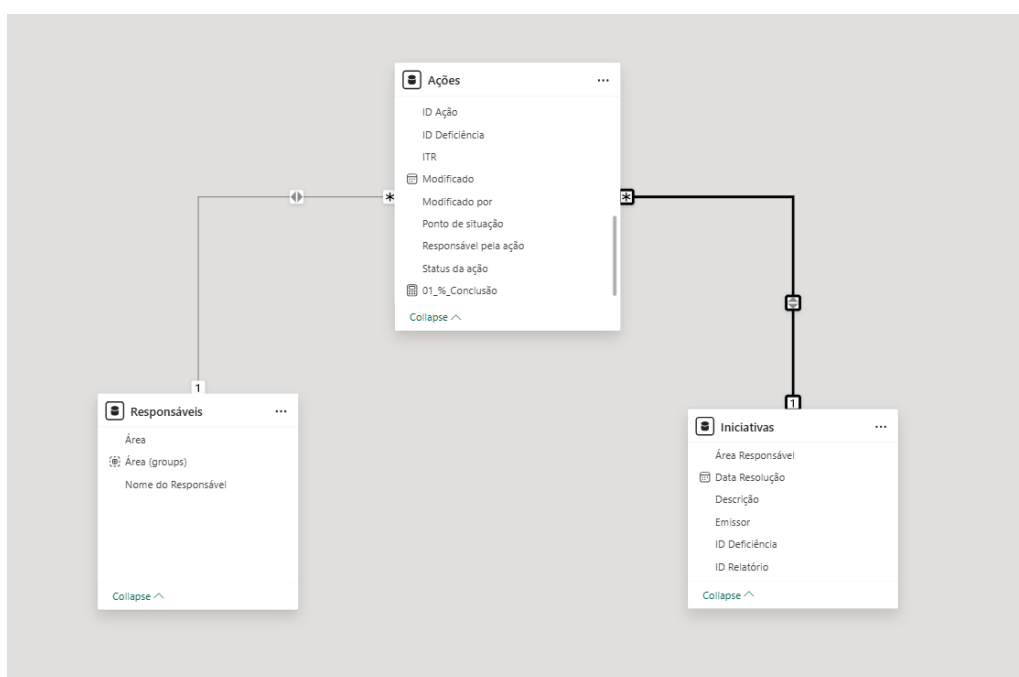


Figura 13 - Esquema relacional Power BI

Na figura 14 é apresentando o dashboard desenvolvido. Os dados e os títulos dos gráficos foram eliminados, por uma questão de confidencialidade dos mesmos. O dashboard apresenta insights sobre a distribuição do status da ação por diversas variáveis de interesse.

No início da folha existe vários filtros que permite ao utilizador filtrar a informação dos gráficos. No fim da folha de dashboard é apresentado um calendário com a respetiva data de início e fim da atividade, bem como o seu status. Por último é apresentando uma tabela com a informação toda agregada. Esta tabela foi solicitada pelas áreas, pois ao selecioná-la, é possível exportar os dados para excel.

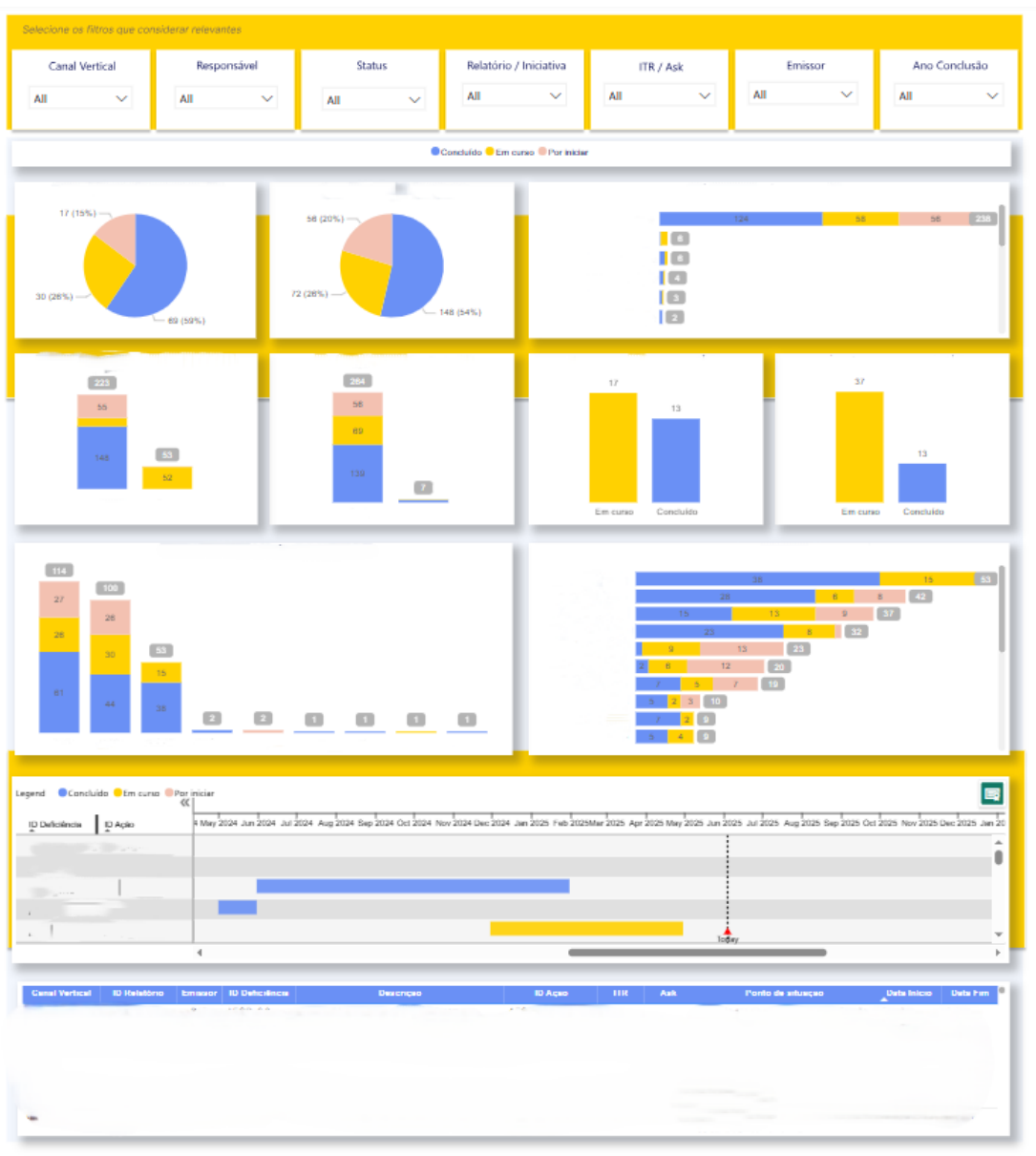


Figura 14 - Dashboard Power BI

4.1.4.6. Microsoft SharePoint

Depois de criar a aplicação em Power Apps e respetivo dashboard em Power BI, foi desenvolvido uma página em SharePoint com o objetivo de centralizar todas as ferramentas utilizadas. Com esta página os utilizadores apenas necessitam aceder a mesma para visualizar, em vez de acederem separadamente as ferramentas, conforme mostrado na figura 15.

Para criar esta página e incorporar as ferramentas, apenas foi necessário publicá-las, e seguidamente retirar o seu link de incorporação em SharePoint disponível no Power

Apps e Power BI. Todas as outras partes mais estéticas da página apenas foi necessário escrever os textos e escolher as respetivas cores e imagens.

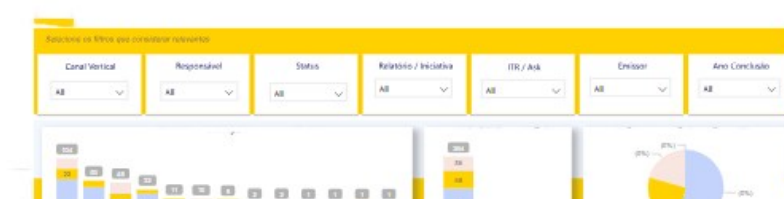
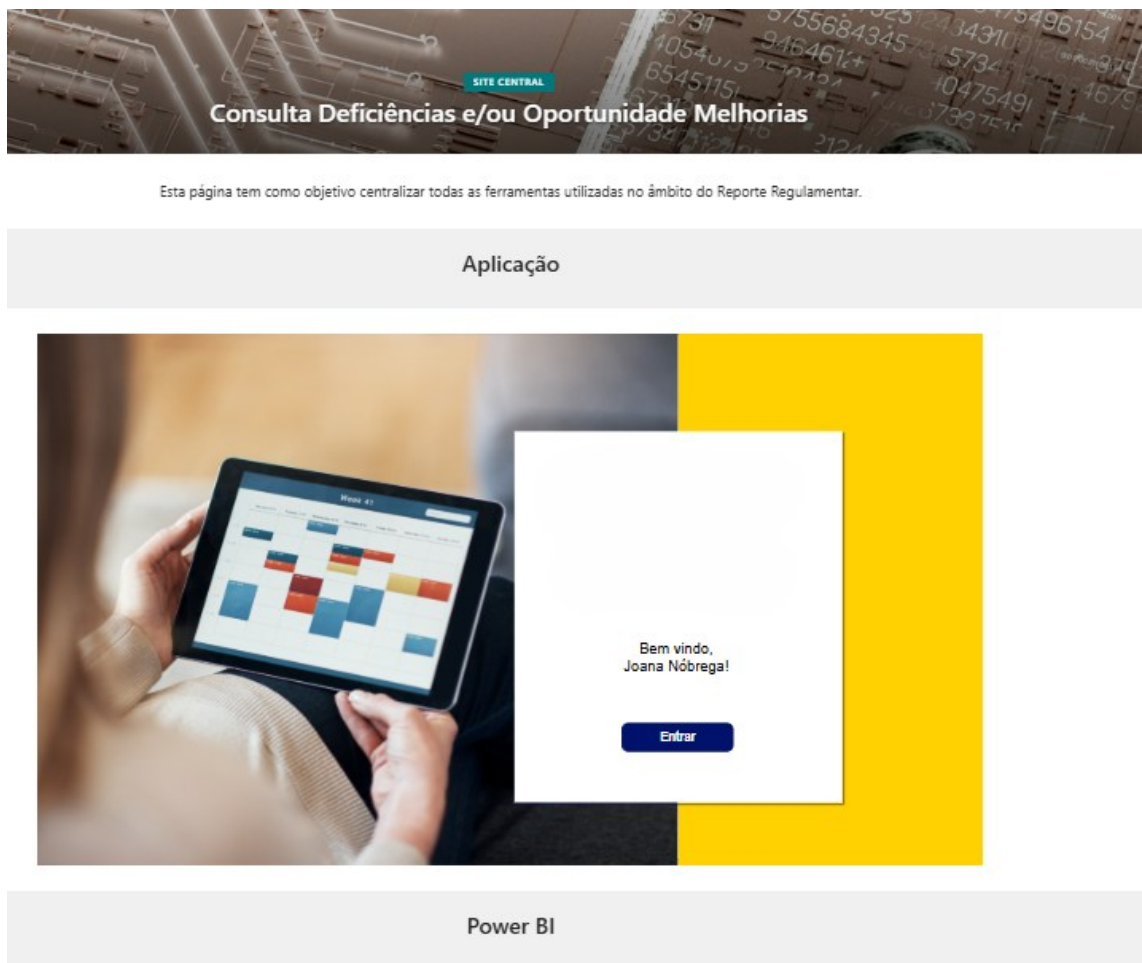


Figura 15 - Página em SharePoint

4.1.5. Testes

Muitos testes foram feitos durante a fase de desenvolvimento da aplicação, ou seja, sempre que alguma página ficava funcional, havia um conjunto de cenários que eram testados na aplicação, desde a inserção de dados que não respeitavam os requisitos previamente definidos até a introdução de dados válidos, sendo sempre verificados o

output de cada ação, com o objetivo de assegurar que todos os componentes individuais estavam devidamente conectados e em funcionamento.

Depois de elaborada a solução, foi realizada uma reunião com alguns dos utilizados finais da solução, durante a qual a aplicação foi apresentada e discutida. Nas sessões foram recolhidos vários feedbacks relevantes relativos aos testes sobre a solução e novas funcionalidades na mesma, originando melhorias posteriores. Este procedimento foi repetido em várias iterações, com novos testes e ajustes progressivos, até a aplicação alcançar uma versão alinhada com todas as necessidades dos utilizadores.

A aplicação encontra-se atualmente em produção, contudo, este tipo de sessões constitui um ciclo iterativo de melhoria contínua, no qual podem surgir novas necessidades ou propostas de desenvolvimento, sendo estas posteriormente analisadas e, sempre que justificadas, incorporadas na solução. Desta forma, a aplicação mantém-se em constante evolução.

5. CONCLUSÕES E LIMITAÇÕES

As ferramentas low-code têm contribuído para a transformação digital nas organizações (Phalake et al., 2023; Sahay et al., 2020). Este tipo de ferramentas permitem aos utilizadores desenvolver soluções que não requeiram código complexo, permitindo que pessoas não detentoras de grande *know-how* técnico em programação, intitulados Citizen Developers (Alsaadi et al., 2021), consigam desenvolver aplicações alinhadas às necessidades da organização. Estas ferramentas contribuem para a redução de processos manuais, a redução de custos, e diminuição dos tempos necessários para desenvolver uma aplicação, quando comparadas com o desenvolvimento tradicional de *software* (Bock & Frank, 2021; Phalake et al., 2023; Sahay et al., 2020). Além disso, as soluções low-code facilitam a manutenção dos serviços e permite às empresas adaptarem-se mais rapidamente perante novos contextos. As ferramentas low-code destacam-se pela facilidade de utilização e adoção (Phalake et al., 2023; Sahay et al., 2020).

Contudo, quando há necessidade de desenvolver uma aplicação mais complexa, a necessidade de código adicional para soluções que precisem de um maior nível de especificidade permanece. A escalabilidade neste tipo de soluções, também constitui um problema, pois muitas plataformas podem não ser uma solução robusta quando o objetivo é criar uma aplicação que requeira trabalhar com um vasto conjunto de dados, nomeadamente *Big Data* (Ihirwe et al., 2020; Luo et al., 2021).

O presente projeto teve como objetivo desenvolver um ecossistema integrado, para responder às necessidades identificadas pela empresa, tendo sido implementada uma solução recorrendo a ferramentas como o Power Apps, Power Automate, Power BI, Lists, Figma e Outlook. Todas estas ferramentas tiveram um papel fundamental no projeto desenvolvido proporcionando a apresentação de uma solução abrangente que receba dados, gira a informação, crie alarmísticas e proporcione a análise e visualização dos mesmos.

A implementação deste ecossistema, sustentado pela aplicação low-code desenvolvida, pelos *workflows* automatizados e pelo *dashboard* de monitorização, demonstrou ser eficaz na resolução do problema inicialmente identificado. A ausência de centralização e visibilidade sobre o estado da informação foi mitigada através da criação de uma plataforma unificada que integra, de forma automática, os dados provenientes dos

diferentes intervenientes do projeto. Este sistema permite não apenas a consolidação e atualização contínua da informação, mas também a sua análise em tempo real, facilitando o acompanhamento do progresso e a identificação antecipada de eventuais desvios no cumprimento dos prazos estabelecidos. O *dashboard* disponibiliza uma visão global e intuitiva sobre o desempenho e evolução do projeto.

Assim, pode-se dizer que a solução desenvolvida atinge plenamente o objetivo proposto e resolve o problema identificado, proporcionado que cada processo seja executado com maior brevidade, aumentando ganhos de tempo por parte dos colaboradores que eram responsáveis por tarefas manuais, diminuindo *bottlenecks* que pudessem existir, melhorando a eficiência e velocidade dos processos de organização.

No entanto, o Power Apps apresenta várias limitações, como por exemplo as seguintes:

- A partilha da aplicação com apenas 50 utilizadores;
- A deteção de erros no código não é intuitiva, sendo difícil perceber o erro;
- Existe vários objetos que estão bloqueados e que não conseguimos aceder;
- Limitações em nível de design.

Relativamente às outras ferramentas, destacam-se as seguintes limitações:

- Microsoft Lists: baixo desempenho em grandes volumes de dados. O Lists consegue fazer face as necessidades como repositório de dados, no entanto, se os dados a utilizar fossem de maior dimensão, esta ferramenta não seria a mais eficiente ou mesmo capaz de dar resposta;
- Microsoft Power Automate: para aceder a todas as funcionalidades disponíveis na aplicação, é necessário obter a versão premium, o que representa um custo elevado.

Algumas destas limitações conseguem ser colmatadas com a utilização da versão premium do Microsoft Power Apps, no entanto, ao elaborar esta aplicação, não foi possível ter acesso a essa versão.

Existe muito espaço para melhorias no desenvolvimento de novas aplicações futuras tais como a utilização de uma base de dados mais robusta, nomeadamente em SQL ou

Dataverse, de forma a proporcionar um maior desempenho a nível de eficiência e tempo de resposta na aplicação.

REFERÊNCIAS

- Alsaadi, H. A., Radain, D. T., Alzahrani, M. M., Alshammari, W. F., Alahmadi, D., & Fakieh, B. (2021). Factors that affect the utilization of low-code development platforms: survey study. *Revista Romana de Informatica si Automatica*, 31(3). <https://doi.org/10.33436/v31i3y202110>
- Ambrose, G., & Harris, P. (2010). Basics Design 08: Design Thinking. Em *Basics Design*.
- Bock, A. C., & Frank, U. (2021). Low-Code Platform. *Business and Information Systems Engineering*, 63(6). <https://doi.org/10.1007/s12599-021-00726-8>
- Bratincevic, J., Taylor, R., & Stone, Z. (2024). *The Low-Code Market Could Approach \$50 Billion By 2028*. Forrester. <https://www.forrester.com/blogs/the-low-code-market-could-approach-50-billion-by-2028/>
- Cameron, R. (2022). *Why Low-Code Development Is A Game Changer*. Forbes. <https://www.forbes.com/councils/forbestechcouncil/2022/07/07/why-low-code-development-is-a-game-changer/>
- Dorst, K. (2011). The core of «design thinking» and its application. *Design Studies*, 32(6). <https://doi.org/10.1016/j.destud.2011.07.006>
- Ebert, C., Henrique, C., & Duarte, C. (2018). *Digital Transformation*.
- Gartner. (sem data). *Business Process Automation (BPA)*. Obtido 8 de Março de 2025, de <https://www.gartner.com/en/information-technology/glossary/bpa-business-process-automation>
- IBM. (sem data). *IBM Robotic Process Automation*.
- Ihirwe, F., Di Ruscio, D., Mazzini, S., Pierini, P., & Pierantonio, A. (2020). Low-code engineering for internet of things: A state of research. *Proceedings - 23rd ACM/IEEE International Conference on Model Driven Engineering Languages and Systems, MODELS-C 2020 - Companion Proceedings*. <https://doi.org/10.1145/3417990.3420208>
- Kholiya, P. S., Kapoor, A., Rana, M., & Bhushan, M. (2021). Intelligent Process Automation: The Future of Digital Transformation. *Proceedings of the 2021 10th*

- International Conference on System Modeling and Advancement in Research Trends, SMART 2021*. <https://doi.org/10.1109/SMART52563.2021.9676222>
- Kraus, S., Jones, P., Kailer, N., Weinmann, A., Chaparro-Banegas, N., & Roig-Tierno, N. (2021). Digital Transformation: An Overview of the Current State of the Art of Research. *SAGE Open*, 11(3). <https://doi.org/10.1177/21582440211047576>
- Kunduru, A. R. (2023). Cloud BPM Application (Appian) Robotic Process Automation Capabilities. *Asian Journal of Research in Computer Science*, 16(3). <https://doi.org/10.9734/ajrcos/2023/v16i3361>
- Luo, Y., Liang, P., Wang, C., Shahin, M., & Zhan, J. (2021). Characteristics and challenges of low-code development: The practitioners perspective. *International Symposium on Empirical Software Engineering and Measurement*. <https://doi.org/10.1145/3475716.3475782>
- Melchert, F., Klesse, M., & Winter, R. (2004). Aligning Process Automation and Business Intelligence to Support Corporate Performance Management. *10th Americas Conference on Information Systems, AMCIS 2004*.
- Microsoft. (2025, Abril 19). *Get started with triggers*.
- Moreira, S., Mamede, H. S., & Santos, A. (2023). Business Process Automation in SMEs. *Lecture Notes in Business Information Processing*, 464 LNBIP. https://doi.org/10.1007/978-3-031-30694-5_31
- Phalake, V. S., Joshi, S. D., Rade, K. A., Phalke, V. S., & Molawade, M. (2023). Optimization for achieving sustainability in low code development platform. *International Journal on Interactive Design and Manufacturing*. <https://doi.org/10.1007/s12008-023-01338-0>
- Radke, A. M., Dang, M. T., & Tan, W. K. A. (2020). Using robotic process automation (RPA) to enhance item master data maintenance process. *Logforum*, 16(1). <https://doi.org/10.17270/J.LOG.2020.380>
- Ramos Alves, F., & Gomes Soares Alcalá, S. (2022). Análise da abordagem LOW-CODE como facilitador da transformação digital em indústrias. *Revista e-TECH: Tecnologias para Competitividade Industrial - ISSN - 1983-1838*, 15(1). <https://doi.org/10.18624/etech.v15i1.1186>

- Reinkemeyer Editor, L. (2020). *Process Mining in Action Principles, Use Cases and Outlook*.
- Richardson, C., Rymer, J. R., Mines, C., Cullen, A., & Whittaker, D. (2014). New Development Platforms Emerge For Customer-Facing Applications. *Forrester: Cambridge, MA, USA*.
- Romão, M., Costa, J., & Costa, C. J. (2019). *2019 14th Iberian Conference on Information Systems and Technologies : 19-22 June 2019, Coimbra, Portugal*. Institute of Electrical and Electronics Engineers.
- Sahay, A., Indamutsa, A., Di Ruscio, D., & Pierantonio, A. (2020). Supporting the understanding and comparison of low-code development platforms. *Proceedings - 46th Euromicro Conference on Software Engineering and Advanced Applications, SEAA 2020*. <https://doi.org/10.1109/SEAA51224.2020.00036>
- Sanchis, R., García-Perales, Ó., Fraile, F., & Poler, R. (2020). Low-code as enabler of digital transformation in manufacturing industry. *Applied Sciences (Switzerland)*, 10(1). <https://doi.org/10.3390/app10010012>
- Schwertner, K. (2017). Digital transformation of business. *Trakia Journal of Science*, 15(Suppl.1). <https://doi.org/10.15547/tjs.2017.s.01.065>
- Sobczak, A. (2020). Developing a robotic process automation management model. *Informatyka Ekonomiczna*, 2019(2). <https://doi.org/10.15611/ie.2019.2.06>
- Talesra, K., & G. S., N. (2021). Low-Code Platform for Application Development. *International Journal of Applied Engineering Research*, 16(5). <https://doi.org/10.37622/ijaer/16.5.2021.346-351>
- Vincent, P., Iijima, K., Driver, M., Wong, J., & Natis, Y. (2019). *Licensed for Distribution Magic Quadrant for Enterprise Low-Code Application Platforms*. <https://www.gartner.com/doc/reprints?id=1-1XQ92DO5&ct=191105&st=sb>
- Weske, M. (2019). Business Process Management Architectures. Em *Business Process Management*. https://doi.org/10.1007/978-3-662-59432-2_8
- Willcocks, L., & Lacity, M. (2016). Service automation robots and the future of work. *Information and Organization*.

Wolniak, R. (2017). SYSTEMY WSPOMAGANIA W INŻYNIERII PRODUKCJI
Inżynieria Systemów Technicznych.

