



Deloitte Touche Tohmatsu Limited & Instituto Superior de Engenharia de Coimbra

---

## **Modernização de Operações de Seguros: Análise Abrangente de uma Abordagem Low-Code Multifacetada para Desenvolvimento Escalável**

---

Almeida, João Filipe  
a2020144466@isec.pt

27 de março de 2024

ISEC Coimbra, Dept. de Engenharia Informática e Sistemas  
Licenciatura em Engenharia Informática - Desenvolvimento de Aplicações

### **Relatório de Estágio**

Supervisionado por:  
Carlos Moraes, Deloitte  
Filipe Sá, ISEC



## Resumo

Este relatório descreve um trabalho onde se aborda a modernização das operações de seguros através da implementação de uma abordagem *low-code*, com foco na plataforma OutSystems. Ao longo do estágio curricular, foram enfrentados desafios significativos, desde a adaptação a novas tecnologias até à integração eficiente de sistemas existentes. A análise abrangente do desenvolvimento *low-code* multifacetado destaca os benefícios e as considerações essenciais ao utilizar esta abordagem no contexto específico das operações de seguros.

Este relatório visa contribuir para a compreensão do impacto do desenvolvimento *low-code* na área de seguros, destacando as vantagens e desvantagens das várias tecnologias, os seus competidores e como se interligam entre si, nomeadamente Azure, MongoDB, Nuxeo e competidores.

Toda a descrição do processo acontecerá num contexto de um estágio curricular cujo propósito foi o avanço académico do estagiário na área de Tecnologias de Informação e cujas tarefas estenderam-se entre a correção e reporte de defeitos da aplicação, a análise e acompanhamento dos utilizadores nos seus incidentes e a execução de terceiras tarefas como outras que tenham sido necessárias ou requisitadas, bem como a tentativa de constante automação e aprimoramento dos processos usados nas resoluções.

**Palavras-chave:** outsystems, *low-code*, seguros, seguradoras, mongodb, azure, nuxio, subscriptor, corretor, contrato, atlas, service center, lifetime, service studio, ServiceNow



# Abstract

This report describes a project that addresses the modernization of insurance operations through the implementation of a low-code approach, focusing on the OutSystems platform. Throughout the academic internship, significant challenges were encountered, ranging from adapting to new technologies to efficiently integrating existing systems. The comprehensive analysis of multifaceted low-code development highlights the benefits and essential considerations when employing this approach in the specific context of insurance operations.

This report aims to contribute to the understanding of the impact of low-code development in the insurance sector, highlighting the advantages and disadvantages of various technologies, their competitors, and how they interconnect with each other, namely Azure, MongoDB, Nuxeo, and competitors.

The entire process description occurs within the context of a curricular internship aimed at advancing the intern's academic knowledge in the field of Information Technologies. Tasks included bug correction and reporting, analysis, and user support in incidents, as well as the execution of third-party tasks as needed or requested, along with ongoing attempts at process automation and improvement.

**Keywords:** outsystems, low-code, insurance, mongodb, azure, nuxio, underwriter, broker, contract, atlas, service center, lifetime, service studio, ServiceNow



## Epígrafe

*“A ciência é mais que um corpo de conhecimento, é uma forma de pensar, uma forma cética de interrogar o universo, com pleno conhecimento da falibilidade humana.”*

Carl Sagan



## Agradecimentos

Quero aproveitar esta oportunidade para expressar a minha sincera gratidão a todos os que contribuíram de alguma forma para o desenvolvimento e conclusão deste projeto.

Gostaria de expandir os meus agradecimentos ao meu orientador na empresa Deloitte, Carlos Morais, pelo valioso suporte, orientação e contribuições fundamentais ao longo deste projeto. O seu excelente acompanhamento, a sua constante e perspicaz avaliação, assim como a gestão dos diferentes membros da equipa com variadas personalidades e capacidades fez-me crescer como profissional e vai sem dúvida deixar um impacto duradouro na minha futura carreira profissional. Sinto-me igualmente endividado ao meu orientador universitário, Filipe Sá, que não só estimo como supervisor cujo conhecimento e proficiência na área de informática de sistemas e imaculado historial na orientação de estágios me permitiram manter um alto rigor e qualidade na elaboração do relatório, e sinto-me bastante privilegiado por ter a oportunidade de aproveitar o seu aconselhamento cujos contributos foram essenciais para o sucesso desta iniciativa.

Estendo os meus agradecimentos a toda a equipa da Deloitte, incluindo os membros da equipa em que fui integrado, João Freitas, Nuno Gaio, Andreia Gomes, Liliana Ferreira, Márcio Costa, Jorge Tavares e Laisa Lamir, pela colaboração, dedicação e trabalho em equipa excepcionais com os quais sempre tive uma relação próxima e acolhedora, amplio os meus agradecimentos ainda aos restantes membros das equipas do projeto com quem sempre me senti à vontade para inquirir acerca de quaisquer dúvidas e sempre estive confortável para partilhar qualquer questão, e sempre me ajudaram no crescimento dentro da empresa e no projeto.

Agradeço ainda a todos na comunidade académica do ISEC, desde professores e docentes até colegas e alunos, pela troca de conhecimentos e experiências enriquecedoras ao longo deste percurso educacional que me deixaram apenas com um grande carinho pela instituição e por todas as experiências que me proporcionaram, nunca esquecerei o meu tempo de grande aprendizagem e desenvolvimento que tive o prazer de passar no ISEC e considero-me bastante sortudo por ter tido esta oportunidade, obrigado a todos!



# Conteúdo

<b>Resumo</b>	<b>iii</b>
<b>Abstract</b>	<b>v</b>
<b>Epígrafe</b>	<b>vii</b>
<b>Agradecimentos</b>	<b>ix</b>
<b>Abreviaturas</b>	<b>xvii</b>
<b>1 Introdução</b>	<b>1</b>
1.1 ISEC . . . . .	1
1.2 Deloitte . . . . .	2
1.3 Contextualização . . . . .	3
1.4 Objetivos . . . . .	4
1.5 Plano de Trabalhos . . . . .	5
1.6 Metodologia de Trabalho . . . . .	5
1.7 Estrutura do Relatório . . . . .	7
<b>2 Formações</b>	<b>9</b>
2.1 Formação de PL/SQL . . . . .	9
2.2 Formação de Java . . . . .	10
2.3 Formação de Excel . . . . .	11
2.4 Formação de SQL e Modelação . . . . .	12
2.5 Formação de Agile . . . . .	12
2.6 Formação de OutSystems . . . . .	15
2.7 Formação de Azure . . . . .	17
2.8 Formação de OWASP . . . . .	18
2.9 Formação de MongoDB . . . . .	19
2.10 Outras formações de curta duração . . . . .	19
<b>3 Conceitos Importantes</b>	<b>21</b>
3.1 Mercado de Seguros de Londres . . . . .	21
3.2 Desenvolvimento <i>Low-Code</i> . . . . .	25
3.3 Terminologia do Projeto . . . . .	29
<b>4 Organização do Projeto e Empresa</b>	<b>33</b>
4.1 Estrutura Organizacional da Deloitte PT . . . . .	33
4.2 Estrutura Organizacional da RIL na Deloitte PT . . . . .	36

<b>5 Infraestrutura Tecnológica do RIL</b>	<b>41</b>
5.1 Integração com Ferramentas na Nuvem . . . . .	41
5.2 APIs . . . . .	43
5.3 Versionamento . . . . .	44
<b>6 Ferramentas e Plataformas usadas no desenvolvimento da RIL</b>	<b>49</b>
6.1 OutSystems . . . . .	49
6.2 ServiceNow . . . . .	58
6.3 Plataforma Atlas . . . . .	61
6.4 Plataforma Azure . . . . .	62
6.5 Plataforma MS Teams . . . . .	64
6.6 Jira & Confluence . . . . .	66
<b>7 Tarefas no Âmbito do Projeto RIL</b>	<b>71</b>
7.1 Resolver <i>Defects</i> . . . . .	71
7.2 Reportar <i>Defects</i> . . . . .	79
7.3 Análise de Incidentes . . . . .	79
7.4 Testes . . . . .	84
7.5 Análise de Processos . . . . .	85
<b>8 Conclusão</b>	<b>95</b>
8.1 Desafios encontrados e melhorias . . . . .	95
8.2 Reflexões Finais . . . . .	96
8.3 Trabalho Futuro . . . . .	97
<b>A Proposta de Estágio</b>	<b>A1</b>
<b>B Mapa de Gantt das Formações do Estágio</b>	<b>B1</b>
<b>C Requisitos Funcionais dos Scripts de Python</b>	<b>C1</b>
<b>D Scripts de Python</b>	<b>D1</b>
D.1 <i>sdcprocesses-ActiveLoops.py</i> . . . . .	D1
D.2 <i>mrcprocesses.py</i> . . . . .	D7
D.3 <i>4VDocumentsprocesses.py</i> . . . . .	D16
D.4 <i>requirements.txt</i> . . . . .	D27

# **Lista de Figuras**

1	Mapa de Gantt das atividades no estágio . . . . .	6
2	Possível solução apresentada de diagrama de casos de uso para o sistema de uma máquina de vendas . . . . .	10
3	Agile VS Waterfall . . . . .	14
4	Scrum Process . . . . .	14
5	Interface de OutSystems - Aplicação criada durante a formação . . . . .	15
6	Maiores mercados de seguros - 2013 . . . . .	23
7	Titanic Placement Slip — 2013 . . . . .	24
8	Maiores razões para usar <i>low-code</i> . . . . .	27
9	Maiores causas de atrasos na entrega de aplicações <i>low-code</i> . . . . .	28
10	Extensão da OutSystems . . . . .	44
11	Usable actions by the API . . . . .	45
12	<i>Deployment Aggregator</i> . . . . .	47
13	Diretriz 1 — Sentido descendente é progresso . . . . .	51
14	Diretriz 2 — Ramificações para a direita . . . . .	51
15	Diretriz 3 — Exceções para a direita . . . . .	52
16	Diretriz 4 — Ciclos . . . . .	52
17	Diretriz 5 — Evitar sobreposições . . . . .	52
18	Diretriz 6 — Alinhar com intenção . . . . .	52
19	Google Trends for Low Code Platforms Worldwide . . . . .	54
20	Mendix UI . . . . .	55
21	Appian UI . . . . .	56
22	Pega UI . . . . .	57
23	Plataforma ServiceNow - Pagina Principal . . . . .	58
24	Plataforma ServiceNow - Incidentes . . . . .	60
25	Plataforma Atlas . . . . .	61
26	Plataforma Azure . . . . .	63
27	Plataforma do Microsoft Teams . . . . .	65
28	Plataforma Jira Software . . . . .	67
29	RIL Next Gen <i>Defect</i> workflow . . . . .	68
30	RIL Next Gen SubTasks workflow . . . . .	68
31	Plataforma do Confluence . . . . .	69
32	Index Out of Bounds - Root of the Problem . . . . .	75
33	Interface de análise de um processo - Service Center . . . . .	85
34	Mapa de Gantt das formações . . . . .	B1



# **Lista de Tabelas**

1	Princípios do manifesto Agile	13
2	Descrição dos ambientes	46
3	Comparação de Plataformas Low-Code	54
4	<i>Defects Trabalhados</i>	72
5	Detalhes do defeito ' <i>Index out of bounds when submitting cancel and replace</i> '	74
6	Detalhes do defeito ' <i>Stamp Visibility   Broker - The UW stamps should be filtered by the company</i> '	77
7	Detalhes da US " <i>List of Stamps - Add a Stamp</i> "	78
8	Defeitos Reportados	80
9	Incidentes Trabalhados	81
10	Incidente INC0064225	82
11	Incidente INC0065686	83
12	Detalhes da US <i>Análise de Processos SDC no Estado "Active - Loop"</i>	88
13	Detalhes da US <i>Análise de Processos GenerateMRCEmailProcess com Utilizadores Inativos</i>	89
14	Detalhes da US <i>Análise de Processos GenerateMRCEmailProcess com Utilizadores Inativos</i>	90
15	Detalhes da US <i>Análise de Processos GenerateMRCEmailProcess com Utilizadores Inativos</i>	90
16	Requisitos funcionais dos scripts para análise de processos	91
17	Requisito funcional <i>Análise de Processos SDC</i>	91
18	Requisito funcional <i>Análise de Utilizadores Inativos em GenerateMRCEmailProcess</i>	C1
19	Requisito funcional <i>Interrupção Controlada do Script</i>	C1
20	Requisito funcional <i>Análise de Documentos em UploadDocuments_V4</i>	C2
21	Requisito funcional <i>Pintar Colunas com Erros de Roxo para Análise Manual</i>	C2



## Abreviaturas

- AD Factory — *Application Development Factory*
- AMS — *Application Managed Services*
- AWS — Amazon Web Services
- BA — *Business Analysts*
- CMMI — *Capability Maturity Model Integration*
- DB — *Data Base*
- FO — *Firm Order*
- GWP — *Gross written premium*
- ISEC — Instituto Superior de Engenharia de Coimbra
- IaaS — *Infrastructure as a Service*
- MF — *Master Facility*
- MQL — *MongoDB Query Language*
- MRC — *Mutual Responsibility Contract*
- MS — Microsoft
- OS — OutSystems
- PL/SQL — *Procedural Language/Structured Query Language*
- PaaS — *Platform as a Service*
- SO — Sistema Operativo
- SQL — *Structured Query Language*
- SaaS — *Software as a Service*
- TI — Tecnologia de Informação
- UI — *User Interface*
- UML — *Unified Modeling Language*

- UMR — *Unique Market Reference*
- URL — Uniform Resource Locator
- US — *User Story*
- UW — *Underwriter*
- VM — *Virtual Machine*(Máquina Virtual)

# 1 Introdução

Este relatório descreve o projeto e atividades desenvolvidas durante o estágio curricular do autor para a conclusão da sua licenciatura realizada no Instituto Politécnico de Coimbra, realizado na Deloitte Touche Tohmatsu Limited. O percurso divide-se em duas fases, uma primeira direcionada à formação e integração do estagiário e a segunda direcionada ao projeto, sua infraestrutura e às diferentes tarefas resolvidas no âmbito deste.

Entre as formações realizadas incluem-se PL/SQL, Java, Excel, SQL e Modelação, Agile e Scrum, OutSystems, Azure, OWASP e MongoDB. Além disso, foram oferecidas formações de curta duração abordando tópicos como Boas Práticas de Desenvolvimento, Banca, Seguros, *Getting Things Done* e *Capability Maturity Model Integration* (CMMI).

A aplicação prática de tecnologias no estágio passou pela utilização de ferramentas e linguagens como Azure, OutSystems, SQL e MongoDB Query Language (MQL) na resolução e análise de *defects*<sup>1</sup> e incidentes. Além disto foram utilizados Excel, Azure e Python no âmbito da análise e reprocessamento de processos do projeto RIL<sup>2</sup>. Foram também realizados numerosos testes com recurso à própria plataforma. Detalhes mais específicos sobre estas experiências e as suas aplicações serão descritos nos capítulos subsequentes.

## 1.1 ISEC

O [Instituto Superior de Engenharia de Coimbra](#) (ISEC) É uma subdivisão do [Instituto Politécnico de Coimbra](#) (IPC) dedicada às engenharias do instituto.

É uma instituição com várias décadas de história, sendo possível traçar relação ao Instituto Industrial e Comercial de Coimbra fundado em 1921[1] fundado com a intenção de atender à crescente necessidade de engenheiros especializados. Foi dividida no Instituto Superior de Engenharia de Coimbra e no Instituto Comercial de Coimbra devido ao Decreto-Lei n.º 830[1] em 1974 que converteu os institutos industriais em institutos superiores de engenharia. Foi apenas em 1988[2] através do Decreto-Lei nº389/88[3] que foi integrado no Ensino Superior Politécnico, consolidando a instituição ainda mais como uma das escolas principais de engenharia em Portugal, continuando até hoje a expandir a sua oferta formativa de modo a se adaptar à indústria bastante volátil.

O ISEC oferece uma vasta gama de cursos a vários níveis académicos tendo CTeSPs, licenciaturas e mestrados e cobrindo vários ramos de engenharia, alguns destes cursos são: Bioengenharia, Engenharia Biomédica, Engenharia Eletrotécnica e de Computadores, Engenharia

---

<sup>1</sup>Uma incongruência identificada entre o funcionamento do programa e as USs (User Stories) definidas. São identificados com uma prioridade e registados na Jira como descrito na Secção 6.6.1.

<sup>2</sup>Nome dado à plataforma e projeto trabalhado através da Deloitte, como explicado em detalhe na Secção 1.3.

Eletromecânica, Engenharia Civil, Engenharia e Gestão Industrial e muitos mais, destingindo-se devido ao seu método prático de aprendizagem e à sua proximidade com o mercado de trabalho, oferecendo bastantes oportunidades de contacto com empresas, muitas vezes sendo intrínseco ao currículo do curso.

A vida no campus é também bastante rica, havendo uma abundância de atividades extra curriculares e associações académicas que contribuem para o desenvolvimento das bastante procuradas no mercado “Soft Skills”, destas atividades fazem parte: A Tuna do ISEC, o EcoCampus do ISEC, a Associação de Estudantes, a Comissão de Estudantes, o Serviço de Apoio à Praxe (SAP), várias instalações e equipas de desporto como a equipa de futebol do IPC, e eventos mais chegados às empresas como a Feira de Engenharia de Coimbra (FENGE), o Fikalab ou o PoliEmpreende, entre outros. Estimulando assim o pensamento crítico e o crescimento social dos alunos, ajudando também na construção de conexões relevantes para o mercado de trabalho.

A própria cidade de Coimbra é bastante acolhedora academicamente sendo frequentemente referida como “A cidade dos estudantes”, é casa da universidade mais antiga de Portugal[4] bem como a mais antiga associação de estudantes do país[5]. Providenciando assim um ambiente académico que incita à criatividade e à exploração.

Em conclusão, o ISEC é uma escola de engenharia bastante prestigiada e procurada por empresas de todo o país, com sólidos princípios de profissionalismo e qualidade capaz de prestar uma educação de excelência e bastante próxima do mercado aos estudantes que por lá passam, tendo tido o privilégio de experienciar em primeira mão esta instituição de ensino de renome.

## 1.2 Deloitte

A empresa foi fundada por William Welch Deloitte em 1845 em Londres, um jovem visionário que começara a trabalhar aos 15 anos como assistente do Administrador do “Bankruptcy Court” onde começou a adquirir a sua experiência na área da consultaria que mais tarde aplicou na empresa. Ao longo da vida de William, a empresa foi bastante bem sucedida e sofreu grandes alterações, isto por ter sido fundada numa época de grande crescimento económico em Londres. No seu primeiro ano, teve quase 100 clientes, muitos deles passando por uma fase no “Bankruptcy Court”. Em 1849, William tornou-se contabilista para a Great Western Railway, uma das primeiras empresas de capital aberto em Londres[6]. Em 1857 teve o seu primeiro parceiro, mudando o nome da empresa para Deloitte and Greenwood e em 1880 foi aberto o primeiro escritório em New York[7]. Deixou a empresa em 1897 com 70 funcionários[6].

Hoje em dia, a Deloitte é uma das maiores consultoras do mundo, sendo frequentemente mencionada como uma das “big four” com outras como a PwC, EY e KPMG[8], dando trabalho a mais de 415 000 pessoas[9].

Até hoje mantém-se uma empresa privada, pelo que as ações não são trocadas publicamente, mas oferece relatórios anuais com estatísticas em relação ao valor da empresa, aos empregados contratados, ao impacto ambiental, etc. Tendo reportado uma receita global de 64.9 mil milhões de US\$ para o ano Fiscal de 2023 com um aumento de 14.9% em moeda local comparado com 2022[10].

A Deloitte oferece uma vasta gama de serviços que são de forma geral divididos entre várias categorias, incluindo: Auditoria e Seguros, Consultoria, Consultoria Financeira, Consultoria em Gestão de Risco, Impostos e Consultoria Jurídica.

Existe na empresa uma forte ética de trabalho, havendo estruturas que se certificam que todos os membros estão a par dela, tendo uma plataforma de e-learnings dos quais alguns são obrigatórios a todos os membros para nos ensinar de forma interativa assuntos como a ética da empresa, os valores, e todos os assuntos internos relevantes.

A segurança dos dados constitui uma grande prioridade para a empresa, fazendo questão que o ambiente de desenvolvimento e comunicação de cada membro esteja atualizado e livre de software de terceiros, tendo por isso a sua própria loja de software interno da qual apenas participa software devidamente examinado e aprovado para uso.

É bastante acolhedora para novos contratados, providenciando os materiais de trabalho, não necessitando assim que os empregados contribuam com qualquer equipamento.

A Deloitte leva muito a sério as formações dos seus contratados, pelo que organiza formações e muitas vezes oferece-se para pagar certificações aos colaboradores. A Deloitte oferece formações relevantes para os novos contratados numa vasta gama de tecnologias preparando os formandos para qualquer projeto em que possam vir a ser introduzidos. No estágio em estudo, houve o privilégio de participar em formações de PL/SQL, Java, Excel, SQL e Modelação, Boas Práticas de Desenvolvimento, Banca, Seguros, Agile/Scrum, Getting Things Done, Capability Maturity Model Integration (CMMI), Testes, OutSystems e Azure.

### 1.3 Contextualização

Devido ao modelo de *outsourcing* da Deloitte e a assuntos de confidencialidade, não é possível mencionar diretamente o nome da empresa para a qual o trabalho foi feito, pelo que, sempre que necessário, este relatório referir-se-á à empresa pelo nome RIL (RiskGuard Insurance Limited).

A RIL é uma nova plataforma eletrónica de negociação para o mercado de seguros de Londres. É o mercado de Londres reconstruído e reinventado para um ambiente digital, visa disseminar valores como a flexibilidade, intuitividade e eficiência, tornando o mercado de Londres um mercado moderno.

No âmbito do estágio, as funções desempenhadas focaram-se na identificação e correção de *defects* e incidentes<sup>3</sup> na plataforma RIL, bem como a realização de testes, o reprocessamento de processos ou a execução de quaisquer outras tarefas pedidas. A necessidade deste estágio surgiu da importância em aprimorar a estabilidade e eficiência da plataforma, e da alta pressão a ser sentida, especialmente na área de triagem, no projeto. Os *defects* e problemas encontrados não só comprometem a integridade do sistema como também afetam a experiência do utilizador. As alterações propostas e o trabalho realizado visam não apenas corrigir estas questões, mas também implementar melhorias que fortalecerão a base digital da RIL para o futuro do mercado de seguros de Londres.

## 1.4 Objetivos

Os objetivos do relatório de estágio poderão ser divididos em duas categorias: os objetivos curriculares, e os objetivos da companhia para o estágio:

### Objetivos Curriculares:

- Promover as capacidades técnicas do aluno nas variadas ferramentas utilizadas, incluindo o ecossistema de OutSystem, Azure, MongoDB e ferramentas de colaboração como o Jira e o Teams;
- Promover o desenvolvimento das “soft-skills” do aluno através do contacto e comunicação constante entre equipas e utilizadores num contexto profissional;
- Dar a oportunidade ao aluno de aplicar os conhecimentos académicos num contexto profissional, aprendendo também como é que um ambiente assim funciona;

### Objetivos para a empresa:

- Analisar, documentar, criar, categorizar, e resolver *defects* do projeto;
- Analisar “Incidents” de utilizadores, falando com eles se necessário, corrigi-los se necessário ou reportando *defects* deles;
- Perceber a logística do projeto, as variadas ferramentas e equipas que nele trabalham para poder delegar certas tarefas, perguntas ou problemas à equipa certa. Desta forma maximizando a eficácia da solução;
- Ganhar conhecimento dos funcionamentos internos da empresa, nomeadamente, através dos e-learnings, o preenchimento mensal da “timesheet” e “expense report”, as variadas plataformas da Deloitte e o objetivo de cada uma delas.

---

<sup>3</sup>Problemas submetidos por utilizadores da aplicação através da ServiceNow [6.2], o utilizador pode marcar a submissão de duas formas: Request ou Incidente, refira à Secção 3.3.

## 1.5 Plano de Trabalhos

O plano de trabalhos previsto ao ser aceite o estágio, tinha objetivos gerais de integração numa equipa e aprendizagem do funcionamento da profissão num contexto empresarial, como se pode ver no Anexo A. Para o projeto RIL, no contexto do estágio, o plano de trabalhos poderá ser dividido em três etapas:

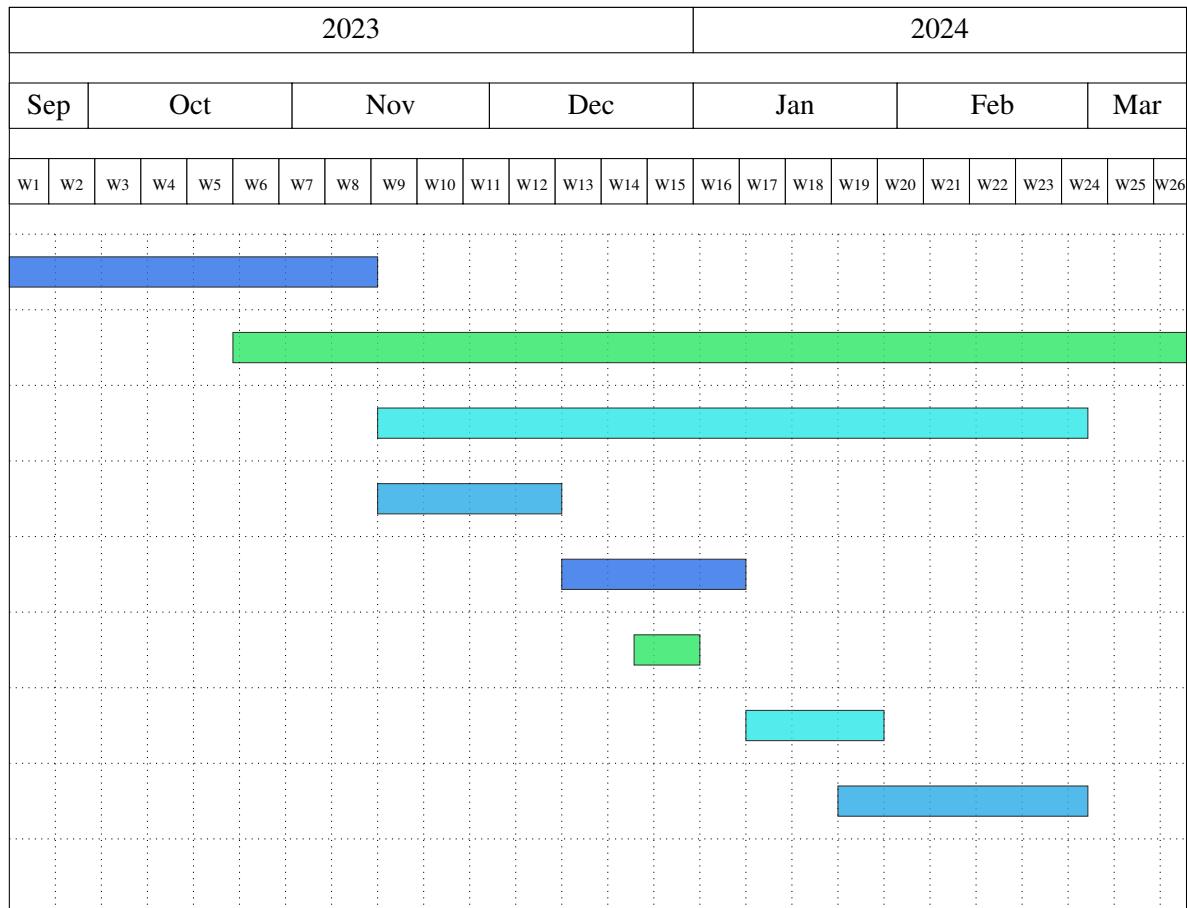
1. Uma fase de formação em tecnologias como OutSystems, MongoDB e Azure como preparação para as tecnologias usadas. Como muitas das fases descritas, não existe uma data precisa que marque o fim desta etapa e o início da próxima, implicando, por isso, que estas formações muitas vezes coincidam com o resto do estágio;
2. Um período de familiarização com os funcionamentos gerais da aplicação e ferramentas de desenvolvimento e gestão de equipas. Esta fase será constante até ao termo do estágio devido à velocidade de desenvolvimento do projeto e ao seu tamanho, pelo que é impossível a um dado momento estar a par de todos as suas complexidades;
3. Tarefas rotativas delegadas às diferentes equipas que mudam a cada duas semanas:
  - **Triagem:** Contacto com problemas reportados diretamente por utilizadores e aplicar uma solução imediata à base de dados ou ao método do utilizador. Se não for possível, é criado um *defect*, isto é, um erro da aplicação para ser analisado e resolvido na lógica da app;
  - **Resolução de Problemas semanais:** Problemas semanais relacionados com logística e outros assuntos difíceis de prever que têm que ser analisados como averiguações de segurança, desempenho, de processos com erros, etc.
  - **Resolução de Defects:** A análise de *defects*, inspeção do código em questão e mudança do mesmo no ambiente correto de forma a resolver o problema.

No final pode-se aferir que as tarefas previstas foram executadas, mas houveram outras não necessariamente planeadas que formaram também uma boa parte do estágio, uma representação fidedigna das tarefas e respetiva calendarização pode-se ver no mapa de Gantt da Figura 1.

## 1.6 Metodologia de Trabalho

As condições e métodos gerais de trabalho na Deloitte são caracterizadas por um ambiente colaborativo e pela utilização de diversas ferramentas e plataformas tecnológicas, nomeadamente:

- **Microsoft Teams e ecossistema:** Utilizado para comunicação e colaboração entre a equipa e membros da Deloitte, facilitando a troca de informações e a coordenação. Muitas ferramentas associadas são também usadas como o calendário, chamadas ou o armazenamento



- 1)  Período exclusivo a formações oferecidas pela Deloitte;
  - 2)  Redação do relatório de estágio;
  - 3)  Projeto RIL;
  - 4)  Adaptação à equipa e resolução de defeitos;
  - 5)  Resolução de incidentes;
  - 6)  Férias de natal;
  - 7)  Execução de testes e reprodução de defeitos;
  - 8)  Reprocessamento de processos de produção.

Figura 1: Mapa de Gantt das atividades no estágio

de dados ou documentos;

- **Outlook:** O cliente de e-mail usado, essencial para comunicações mais importantes ou que devem ser feitas a um certo número de indivíduos;
- **Jira (Confluence):** O Jira, é utilizado para o acompanhamento de projetos, documentação e colaboração em tempo real, funciona em conjunto com o Confluence para organizar uma metodologia ágil Scrum na gestão do projeto;
- **Ferramentas de desenvolvimento** e auxílio ao desenvolvimento como MongoDB, Azure, Oustystems e todo o seu ecossistema.

Cada trabalhador tem acesso ao seu próprio computador pessoal para o trabalho com altos padrões de segurança através de software e regras de uso como a proibição da instalação de qualquer software que não seja do próprio software center da Deloitte, ou a proibição de retirar de dados do computador, por métodos como USB's, garantindo assim uma forte segurança dos dados de trabalho.

É aconselhada a ida da equipa ao escritório pelo menos uma vez por semana, conduzindo a uma interação mais forte entre a equipa e uma relação intergrupal mais saudável.

Todas as equipas participam diariamente em reuniões Scrum de poucos minutos, dando assim a oportunidade de mais facilmente fazer atualizações de condições gerais (consulte secção 2.5.1 para mais informações sobre este método). De seguida há reuniões diárias entre os membros de cada equipa, proporcionando a oportunidade de alinhar as tarefas do dia, discutir o que foi feito no dia anterior, tirar dúvidas, e discutir quaisquer outros tópicos relevantes para a equipa conjunta.

## 1.7 Estrutura do Relatório

O relatório encontra-se organizado de acordo com os seguintes capítulos:

1. **Introdução:** Esta secção apresenta conceitos gerais relacionados ao projeto, destacando as implicações iniciais;
2. **Formações:** São analisadas as formações abordadas no estágio, delineando a preparação e aquisição de conhecimentos adquiridos;
3. **Conceitos Importantes:** Nesta secção, são discutidos os conceitos essenciais no entendimento e compreensão do projeto;
4. **Organização do Projeto e Empresa:** Explora-se a estrutura organizacional dentro da Deloitte, explorando as diferentes dinâmicas de trabalho no contexto da empresa e do

projeto trabalhado;

5. **Infraestrutura Tecnológica do RIL:** Analisa-se a infraestrutura tecnológica envolvida no projeto, fornecendo uma visão mais aprofundada das tecnologias utilizadas e das formas como se interligam;
6. **Ferramentas e Plataformas usadas no desenvolvimento da RIL:** Detalham-se as ferramentas e plataformas utilizadas durante o estágio, destacando as suas vantagens, desvantagens e como foram integradas no projeto e entre si;
7. **Tarefas no Âmbito do Projeto RIL:** Descrevem-se e pormenorizam-se as tarefas específicas trabalhadas, sublinhando-se o processo da sua resolução;
8. **Conclusão:** Nesta secção, são apresentadas as conclusões finais, destacando as principais aprendizagens e realizações do estágio.

## 2 Formações

Neste capítulo será feita uma análise breve de cada uma das tecnologias ou temas abordados durante as formações recebidas no início do estágio. Cada secção a seguir abordará uma tecnologia específica dada numa formação, apresentando uma visão geral, as principais características, a importância e o impacto que cada uma delas pode ter para o desenvolvimento profissional dos profissionais em contexto empresarial.

Existe um caminho formativo bem definido para os novos contratados, comunicado através do DDC Academy (Deloitte Delivery Center) e do CURA, uma plataforma interna de formações onde nos foram passadas grande parte dos materiais e formações.

Enquanto muitas das formações focam no aspeto prático das tecnologias, este capítulo será aproveitado para aprofundá-las e expor quaisquer mais informações relevantes.

O estagiário começou o estágio um pouco depois da data inicialmente acordada por pedido, pelo que algumas das formações e atividades que os restantes recém-contratados comparticiparam na primeira metade do mês de setembro não foram aproveitadas pelo estagiário, na Figura 34 no Anexo B está representado, através de um mapa de Gantt, o tempo dedicado a cada formação no período inicial do estágio, a primeira barra, começa antes do gráfico em si, pois esta formação estava já em decurso antes do começo do estágio, pelo que apenas parte da formação pode ser aproveitada.

### 2.1 Formação de PL/SQL

*Procedural Language/Structured Query Language*, mais conhecido por PL/SQL, é uma linguagem de programação criada pela Oracle para gestão das bases de dados Oracle. Foi criada de forma a integrar declarações SQL na sua sintaxe. Durante a execução, o PL/SQL e o SQL correm no mesmo processo do servidor, otimizando a sua eficiência[11].

O PL/SQL combina a flexibilidade da programação procedural com a capacidade de consulta de dados de SQL, possibilitando a utilização de uma panóplia de conceitos típicos de programação diretamente com SQL, como por exemplo:

- **Triggers de Base de Dados:** Ações ou procedimentos automáticos desencadeados por eventos específicos ou alterações de dados na base de dados;
- **Funções:** O PL/SQL suporta a criação de funções, permitindo menos repetição e mais flexibilidade na organização de dados e código;
- **Procedures & Function Overloading:** Suporta o conceito de criar múltiplas versões de procedimentos ou funções com o mesmo nome, mas diferentes parâmetros, permitindo

flexibilidade e reutilização de código;

- **Exceções:** Como muitas linguagens procedimentais, suporta também o lançamento e tratamento de exceções.

Em suma, o PL/SQL é uma extensão às bases de dados Oracle SQL, e facilita a interação com estas através de uma sintaxe mais familiar para quem está familiarizado com programação procedural, oferecendo ao mesmo tempo, compatibilidade com a sintaxe do SQL[12].

## 2.2 Formação de Java

Java é uma linguagem de programação amplamente utilizada no desenvolvimento de aplicações empresariais, aplicações móveis, sistemas integrados e muito mais. A sua portabilidade, orientação a objetos e robustez fazem dela uma escolha popular para uma variedade de aplicações.

Foram exploradas as maneiras de planeamento de código mais usadas na indústria como o diagrama de casos de uso como o visto na Figura 2, este diagrama identifica as diferentes interações entre atores externos e o sistema, destacando os principais casos de uso e as suas relações. Ajuda a visualizar as funcionalidades que o sistema oferece do ponto de vista do utilizador e fornece uma visão geral da interação entre os atores e os casos de uso.

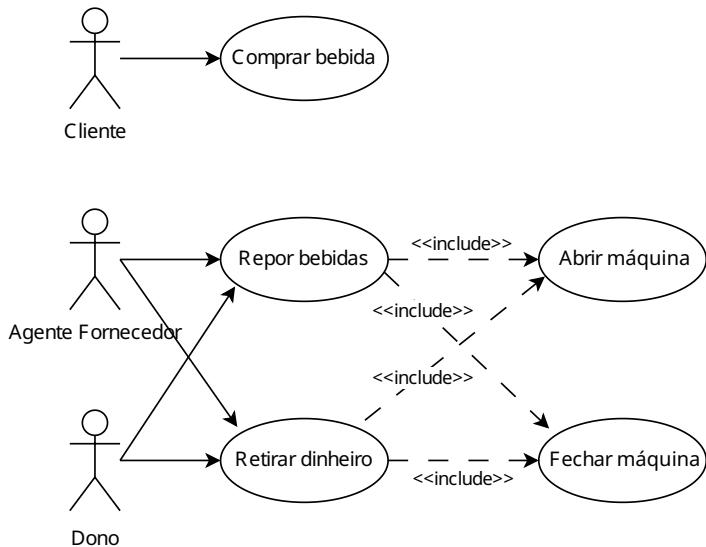


Figura 2: Possível solução apresentada de diagrama de casos de uso para o sistema de uma máquina de vendas

Foram ainda apresentados outros tipos de diagramas tipicamente usados no planeamento de aplicações, nomeadamente:

- **Diagrama de Estados (UML):** Descreve os diferentes estados que um objeto ou sistema pode assumir, destacando as transições entre esses estados e as ações associadas a cada um;
- **Diagrama de Sequências:** Modela a interação entre objetos num sistema em contexto temporal, exibindo a ordem das mensagens trocadas e facilitando a compreensão das dinâmicas do sistema;
- **Diagrama de Classes:** Um diagrama mais próximo da implementação, pois representa a estrutura do sistema orientado a objetos, mostrando classes, atributos, métodos e as relações entre as classes.

Foram ainda cobertos os conceitos fundamentais de programação orientada a objetos bem como tópicos avançados como a manipulação de exceções, hierarquia, polimorfismo, e estruturas de dados.

## 2.3 Formação de Excel

O Microsoft Excel é um programa da suíte Office 365 frequentemente utilizado para armazenar, organizar e manipular dados. É frequentemente utilizado em análise de dados ou criações de visualizações sendo muito usado no contexto da empresa, tornando a capacidade de o utilizar uma aptidão bastante favorável e procurada.

No entanto, esta formação acabou por tocar numa variedade de tópicos mais globais sobre boas práticas na empresa, tocando em tópicos como a forma de se apresentar e lidar com clientes, os cuidados a ter com a partilha de informação e o seu manuseamento, os cuidados a ter com a associação da marca da empresa às nossas vidas pessoais, pontualidade e mesmo a forma de resolver problemas e como a melhor contribuição para um projeto às vezes não precisa de ser técnica.

Existem mais de 400 funções no Excel, permitindo uma interação entre dados bastante flexível, possibilitando a criação de cálculos e até gráficos e visualizações que normalmente não se associam a um programa de spreadsheets. Foram estudadas várias funções, desde a SUM, que simplesmente soma valores de duas células, até VLOOKUP, utilizada para procurar um valor específico na primeira coluna de um conjunto de dados e retornar o valor correspondente de uma coluna diferente na mesma linha. A seguinte lista não exaustiva contempla as funções estudadas durante o percurso:

- SUM, SUMIF, RANGE, CRITERIA, SUMIF, SUMIFS, COUNT, COUNTA, COUNTIF, AVERAGE, MAX, MIN, VLOOKUP, HLOOKUP, XLOOKUP, DCOUNT, DSUM, INDEX, ROUND, ROUNDUP, ROUNDDOWN, LEFT, RIGHT, MID, SEARCH, TRIM, IF, ISNA, MATCH, ISBLANK, ISERROR.

## 2.4 Formação de SQL e Modelação

Dados estão em todo o tipo de aplicações, das pequenas às maiores, e estas aplicações usam bases de dados para guardar toda essa informação. A forma mais comum de guardar informação são bases de dados relacionais, constituídas por tabelas de dados que se podem relacionar entre si[13]. A linguagem mais amplamente usada para consultar estas bases de dados é SQL, sendo usada por uma variedade de serviços de gestão de bases de dados como Oracle, MySQL, PostgreSQL, Microsoft SQL Server[14].

Durante a formação foram abordados todos os temas gerais da linguagem de gestão de dados, focando as seguintes características:

- **Pesquisas Básicas:** As pesquisas mais simples de SQL para obter dados das tabelas, utilizando o SELECT, FROM e WHERE;
- **Ordenação e Agrupamento:** Usar cláusulas como ORDER BY e GROUP BY para organizar e agrupar os resultados de uma consulta SQL com base em critérios específicos;
- **Funções de Agregação:** Envolve o uso de funções como SUM, AVG, MIN e MAX para realizar cálculos em conjuntos de dados, geralmente em combinação com a cláusula GROUP BY;
- **Junções (Joins):** Usa-se para combinar duas ou mais tabelas usando as cláusulas JOIN, INNER JOIN, LEFT JOIN, RIGHT JOIN ou FULL JOIN e foram exemplificadas e visualizadas as diferenças entre cada caso;
- **Subconsultas:** Envolve o uso de consultas dentro de outras consultas, utilizando uma segunda cláusula SELECT na pesquisa;
- **Modificação de Dados:** Utilizou-se INSERT, UPDATE, DELETE e COMMIT para modificar e manipular os dados em tabelas específicas, ou seja, operações CRUD<sup>4</sup>.

## 2.5 Formação de Agile

O Agile é uma abordagem de gestão de projetos e desenvolvimento com foco na flexibilidade, colaboração e satisfação do cliente. Distingue-se de outras abordagens pelo desenvolvimento iterativo, entrega incremental e capacidade de adaptação às mudanças.

Este método diferencia-se do método tradicional Waterfall, linear com fases sequenciais, por ser iterativo. Enquanto uma abordagem Waterfall passará pelas fases de Especificação, Design, Desenvolvimento, Testes e Distribuição de forma sequencial sem avançar para um sem o

---

<sup>4</sup>CRUD (*Create, Read, Update, Delete*) é um acrônimo para formas de se interagir com informação guardada.

último estar completo, a abordagem Agile passará por cada fase simultaneamente, entregando um resultado cada vez mais aprimorado no final de cada ciclo, como representado na Figura 3.

Foram transmitidos os princípios do manifesto Agile, os quais estão delineados na Tabela 1.

Tabela 1: Princípios do manifesto Agile

Fonte: [15]

- A nossa maior prioridade é satisfazer o cliente via a entrega contínua e antecipada do software.
- Entregar software funcional frequentemente, de algumas semanas a alguns meses, com preferência no menor prazo.
- Construir projetos com indivíduos motivados. Oferecer-lhes o ambiente e suporte necessários, e confiar neles para fazer um bom trabalho.
- Em intervalos regulares, a equipa reflete sobre como se tornar mais eficaz, ajustando o seu comportamento.
- Atenção contínua à excelência técnica e bom design melhora a agilidade.
- As melhores arquiteturas, requisitos e designs surgem de equipas organizadas por si mesmas.
- Aceitar mudanças nos requisitos, mesmo no final do desenvolvimento. Processos ágeis aproveitam a flexibilidade das mudanças frequentes em prol vantagem competitiva do cliente.
- Pessoas de negócios e programadores devem trabalhar juntos diariamente ao longo do projeto.
- O método mais eficiente e eficaz de transmitir informações para e entre uma equipa de desenvolvimento é a conversa presencial.
- Processos ágeis promovem o desenvolvimento sustentável. Os patrocinadores, programadores e utilizadores devem ser capazes de manter um ritmo constante indefinidamente.
- Simplicidade — a arte de maximizar a quantidade de trabalho não feito — é essencial.
- Software a funcionar é a principal medida de progresso.

O método ideal dependerá da do projeto, um método Waterfall é mais indicado para projetos menores e previsíveis, enquanto Agile é indicada para projetos mais avançados.

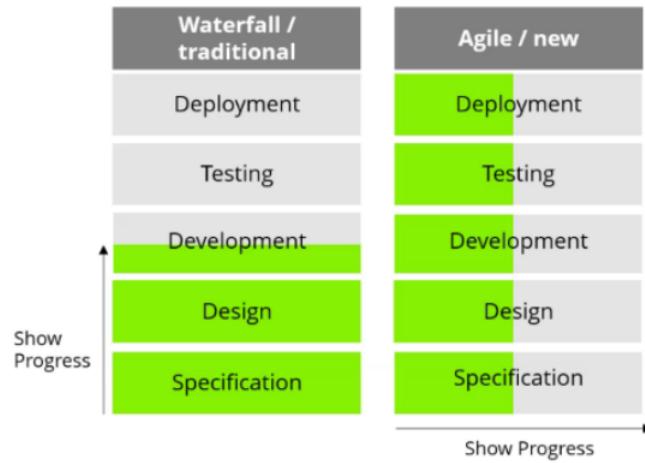


Figura 3: Agile VS Waterfall

Fonte: Deloitte's Delivery Center

### 2.5.1 Framework Scrum

Scrum é um framework de Agile, especifica em detalhe como organizar o desenvolvimento, definindo papéis para as pessoas das equipes e descrevendo como aplicar o processo ágil, cuja representação gráfica apresenta-se na Figura 4.

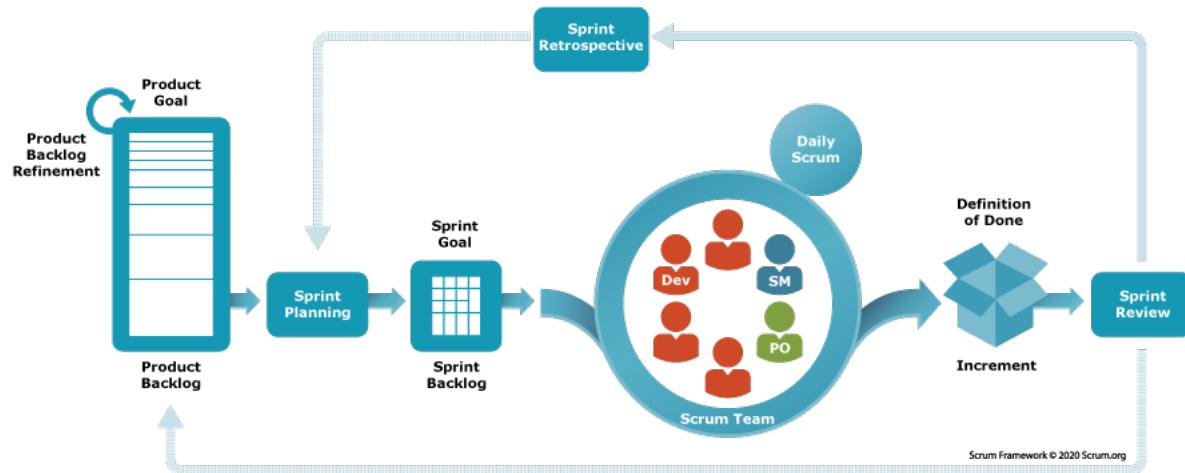


Figura 4: Scrum Process

Fonte: [16]

O Scrum requer um ambiente onde:

- Os Incrementos do trabalho são entregues em ciclos curtos de um mês ou menos, chamados *sprints*. O feedback contínuo ocorre durante a *sprint*, permitindo assim a inspeção e adaptação do processo e do que será entregue;
- A Equipa Scrum possui um *Scrum Master*, um *Product Owner* e Programadores, estes são responsáveis por transformar o trabalho planeado para a *sprint* em algo apresentável;
- A Equipa Scrum e outros membros da organização, utilizadores e clientes conhecidos (*stakeholders*), inspecionam os resultados da *sprint* e ajustam devidamente o planeamento para a próxima.[16].

## 2.6 Formação de OutSystems

OutSystems é uma plataforma *low-code*<sup>5</sup> para desenvolvimento, distribuição e manutenção de aplicações de forma eficiente. Com uma interface visual, como visível na Figura 5, permite a criação de aplicações usando padrões bastante utilizados de forma rápida e eficaz, estando otimizado para a criação e a manutenção destas interfaces facilmente, não deixando de permitir a criação de interfaces e relações entre dados mais avançadas.

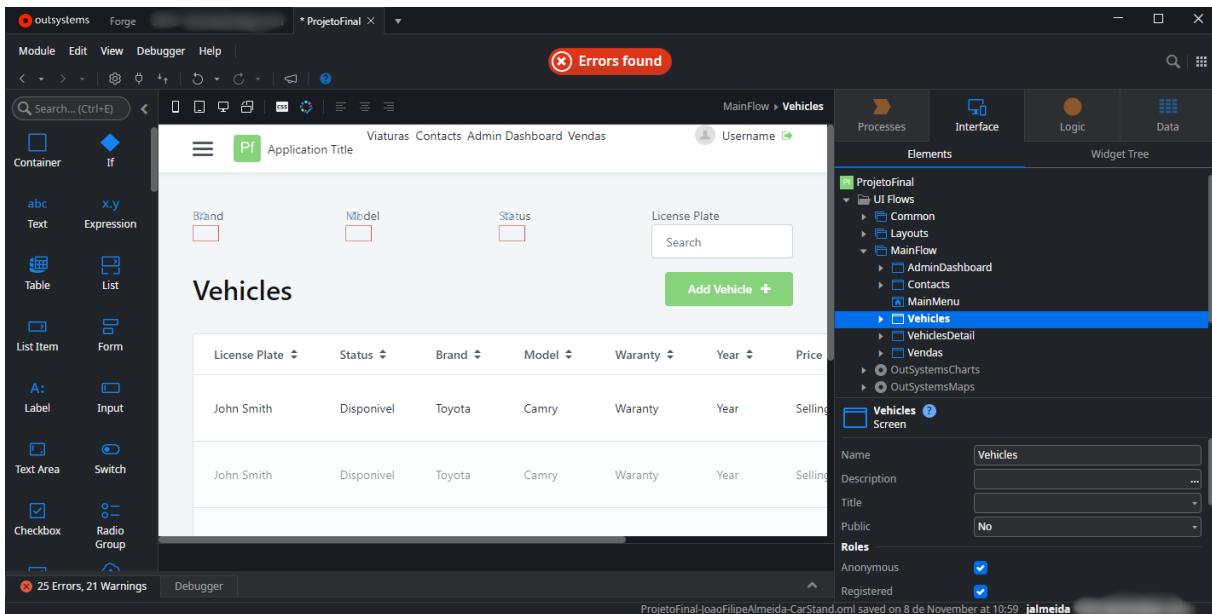


Figura 5: Interface de OutSystems - Aplicação criada durante a formação

Fonte: Plataforma Interna da OutSystems

<sup>5</sup>*Low-code* é uma abordagem de desenvolvimento de software que minimiza a necessidade de programação manual, utilizando plataformas com interfaces gráficas e componentes reutilizáveis. Este método de desenvolvimento acelera o processo permitindo também a participação de profissionais menos experientes em programação.

Para atingir o objetivo de ser uma plataforma visual que cubra o processo todo desde o desenvolvimento à publicação, OutSystems tem que oferecer um conjunto de ferramentas que se interliguem entre si, estas ferramentas são as seguintes:

- **Ferramentas de Desenvolvimento:**

- **Service Studio:** Ambiente visual de desenvolvimento para aplicações Web e Mobile. Quando conectado pode publicar para o “*Platform Server*” (servidores que compilam, publicam, gerem, executam e monitorizam aplicações.) Cada versão destas aplicações vai ser guardada na “*Platform Data Database*”;
- **Integration Studio:** Ambiente de desenvolvimento para criar extensões para a plataforma. Permite a criação de elementos dentro do Service Studio para interagir com recursos externos como código em C# ou bases de dados.

- **Ferramentas de Administração e Operação:**

- **Service Center:** Consola web de administração e gestão do servidor da plataforma. Inspecionar logs gerados pelo ambiente ou pelas aplicações, ou configurar definições do ambiente;
- **LifeTime:** Permite a gestão de todo o ciclo de vida da aplicação em vários ambientes, como, por exemplo, o ambiente de desenvolvimento, o ambiente de qualidade ou o ambiente de produção. É possível gerir as permissões de cada utilizador ou equipa, ver os ambientes e configurá-los ou a sua ordem no ciclo, analisar o desempenho das aplicações, ou efetuar *deploys*<sup>6</sup> de código entre ambientes.
- **OutSystems Forge:** Fonte de componentes para descarregar de forma a acelerar o desenvolvimento;
- **Comunidade:** Fóruns para troca de dicas, colocação de perguntas e obtenção de assistência[17].

Foram também explicados os conceitos mais importantes da plataforma, incluindo a aba da interface, a aba da lógica, a aba dos dados, aggregators, forms e validações, blocos e eventos, relações entre dados, *roles*, Debugging. Para informação mais detalhada da plataforma, por favor refira à Secção 6.1.

---

<sup>6</sup>Passagem de código de um ambiente para outro de acordo com o *Deployment Aggregator*.

## 2.7 Formação de Azure

A Azure é uma plataforma de computação da Microsoft, esta oferece uma panóplia de serviços Cloud para correr aplicações, servidores ou código na cloud.

Existem três níveis distinguíveis de infraestrutura oferecidos:

- **Infrastructure as a Service (IaaS):** IaaS é ideal quando o cliente quer manter controlo total sobre o sistema operativo e aplicações, trata-se de uma infraestrutura onde se virtualiza o hardware e o sistema operativo, deixando à responsabilidade do cliente manter o SO e as aplicações usadas atualizadas;
- **Platform as a Service (PaaS):** Em PaaS a plataforma de desenvolvimento e de publicação é fornecida, não é necessário gerir o sistema ou as aplicações subjacentes, mas o cliente mantém acesso ao código das aplicações que desenvolve, sendo oferecidas uma multitudde de linguagens para as programar;
- **Software as a Service (SaaS):** Em SaaS a única responsabilidade do cliente é a gestão das contas e dos seus credenciais, SaaS refere-se ao fornecimento de aplicações já desenvolvidas para um cliente ou empresa como o Outlook ou o Office.

Dentro destes níveis de infraestrutura, a Microsoft disponibiliza uma variedade de serviços projetados para atender diversas necessidades, desde garantir a redundância de dados até facilitar a transferência de dados entre servidores locais, e a nuvem de Azure. Alguns destes serviços incluem:

- **Azure Blob Storage:** Armazenamento de objetos altamente escalável para dados não estruturados, como imagens e vídeos;
- **Azure Virtual Network:** Permite a criação de redes privadas na nuvem para conectar recursos Azure e estender a rede local;
- **Azure SQL Database:** Um serviço de bases de dados relacionais, proporcionando escalabilidade e segurança;
- **Azure ExpressRoute:** Estabelece conexões privadas e dedicadas à Azure, sem passar pela Internet, permitindo assim a transferência de dados rápida e seguramente;
- **Azure Backup:** Oferece soluções abrangentes de backup para proteger dados críticos, máquinas virtuais e aplicações;
- **Azure Logic Apps:** Permite a automação de fluxos de trabalho utilizando um designer visual permite a integração de serviços para criar aplicações poderosas;

- **Azure Cognitive Services:** Dezenas de serviços baseados em modelos de inteligência artificial oferecidos para integrar com as aplicações como por exemplo: Language cognitive, Speech cognitive, Vision cognitive, Decision cognitive.

Estes são apenas alguns exemplos dos infinidáveis serviços disponíveis na plataforma Azure. Para informações mais detalhadas sobre a plataforma, por favor refira à Secção [6.4](#).

## 2.8 Formação de OWASP

A OWASP, ou *Open Web Application Security Project*, é uma organização sem fins lucrativos dedicada a melhorar a segurança de software, especialmente no contexto de aplicações web. Foi fundada em 2001 e visa ajudar organizações a criar, desenvolver, obter, operar e manter aplicações que podem ser confiadas. [\[18\]](#)

Uma das contribuições notáveis da OWASP é o **OWASP Top Ten**. Esta é uma lista regularmente atualizada dos riscos de segurança mais críticos em aplicações web. A lista ajuda programadores, profissionais de segurança e organizações a priorizarem os seus esforços de forma a melhorar a segurança das aplicações web.

As vulnerabilidades do *OWASP Top ten* como especificadas em 2021 são as seguintes:

- **Controlo de Acesso Quebrado:** Problemas relacionados à má configuração ou implementação inadequada de controlos de acesso, permitindo a utilizadores não autorizados aceder a recursos protegidos;
- **Falhas Criptográficas:** Erros no uso de técnicas criptográficas, como encriptação e assinaturas digitais, que podem levar à exposição de dados sensíveis ou comprometer a segurança do sistema;
- **Injeção:** Vulnerabilidades que surgem quando dados não confiáveis são incorporados em comandos ou consultas, podendo levar à execução não autorizada de código;
- **Design Inseguro:** Riscos associados a falhas de design nas aplicações, refere-se aos riscos que não podem ser corrigidos com a implementação, salientando a importância de um planeamento adequado;
- **Configuração de Segurança:** Problemas decorrentes de configurações inadequadas ou em sintaxe errónea, especialmente em ambientes altamente configuráveis, que podem resultar em exposição não intencional de informações sensíveis;
- **Componentes Vulneráveis e Desatualizados:** Riscos relacionados ao uso de componentes (bibliotecas, frameworks) com vulnerabilidades conhecidas;

- **Falhas de Identificação e Autenticação:** Problemas de autenticação e identificação que podem permitir a utilizadores não autorizados aceder a contas protegidas ou obter privilégios indevidos;
- **Falhas de Integridade de Software e Dados:** Riscos associados a fazer suposições sobre integridade, especialmente relacionadas a atualizações de software e na utilização de extensões ou livrarias de terceiros;
- **Falhas de Registo e Monitorização de Segurança:** Desafios em garantir uma monitorização adequada e registo de eventos de segurança, o que pode afetar a visibilidade, alertas de incidentes e atividades de utilizadores;
- **Falsificação de Pedido do Lado do Servidor:** Situações em que um atacante pode induzir o servidor a fazer pedidos não autorizados em nome do utilizador[19].

Em suma, a OWASP é uma grande ajuda para qualquer empresa para manter um ambiente de aplicações seguras e confiáveis. A comunidade da OWASP é também bastante ativa e é ela que faz o projeto possível, contribuindo ativamente para identificar e documentar vulnerabilidades críticas.

## 2.9 Formação de MongoDB

O MongoDB é um sistema de gestão de bases de dados NoSQL amplamente utilizado, foi projetado para oferecer flexibilidade e escalabilidade em ambientes de desenvolvimento modernos. Ao contrário de Bases de Dados relacionais, o MongoDB adota uma abordagem baseada em documentos, armazenando dados em formato BSON (Binary JSON). A sua estrutura distribuída permite a manipulação eficiente de grandes volumes de dados, enquanto consultas mais complexas são facilitadas por índices e suporte a linguagens de consulta avançada. Além disso, o MongoDB oferece recursos de replicação para garantir alta disponibilidade e tolerância a falhas, sendo uma escolha popular para aplicações que requerem agilidade e desempenho em ambientes dinâmicos[20].

A plataforma usada para interação com estes dados através da nuvem é o Atlas, refira à Secção 5.1.2 para mais informações sobre a plataforma.

## 2.10 Outras formações de curta duração

De seguida apresento uma lista não exaustiva de outras formações de curta duração que muitas vezes se limitavam apenas a uma sessão ou reunião, este tipo de formações são frequentemente oferecidas sem caráter obrigatório aos empregados da empresa, e visam aprimorar as habilidades dos funcionários em diversas áreas, proporcionando-lhes uma vantagem competitiva, incenti-

vando a versatilidade e ambientes em que os profissionais estão constantemente atualizados e no auge de sua eficiência:

- **Boas Práticas de Desenvolvimento:** Nesta formação foram abordados vários temas relacionados à criação de código legível e fácil de manter, foram abordados vários excertos analisados e simplificados colocando excertos relevantes em funções à parte, usando nomes de variáveis e funções descritivos, analisando a utilidade dos comentários, e analisando cada caso específico de forma a oferecer uma visão e capacidade geral de como fazer código profissionalmente;
- **Banca:** Focando-se no setor bancário, um setor onde se situam muitos dos clientes da companhia, foram estudados vários conceitos da área, e as formas como bancos se mantém rentáveis. Existem operações ativas, em que o banco se compromete a devolver o dinheiro ao cliente, como um depósito, é destas operações que o banco ganha o dinheiro que utiliza nas operações passivas, em que o cliente tem que pagar ao banco, por exemplo, um empréstimo, que terá juros;

Foram analisados os riscos associados, incluindo: Risco de Conformidade, Risco Operacional, Risco de Sistemas de Informação, Risco de Estratégia, Risco Reputacional;

- **Seguros:** Focando-se no setor dos seguros, área também de muitos clientes da empresa, esta formação focou nos vários tipos de contratos que um indivíduo pode estabelecer com estas empresas e nas diferenças entre eles;
- **Getting Things Done:** Esta formação veio a otimizar a gestão do tempo e aumentar a produtividade dos formados, proporcionando técnicas e ferramentas para a organização eficiente de tarefas pessoais e profissionais;
- **Capability Maturity Model Integration (CMMI):** CMMI é um modelo que ajuda as organizações a otimizar a melhoria de processos e incentiva a comportamentos produtivos e eficientes que reduzem os riscos no desenvolvimento de software, produtos e serviços, definindo os seguintes níveis de maturidade: *Initial, Managed, Defined, Quantitatively Managed, e Optimizing*[21].

### 3 Conceitos Importantes

Neste capítulo alguns conceitos importantes para compreender o projeto serão elaborados de forma a providenciar uma base forte para o resto do relatório.

#### 3.1 Mercado de Seguros de Londres

Neste capítulo, far-se-á uma análise abrangente do mercado de seguros de Londres, de um ponto de vista histórico, contemporâneo e dos mecanismos intrínsecos que formam as seguradoras na capital britânica. Ao explorar a história e os funcionamentos do setor espera-se estabelecer as bases para uma análise detalhada do Projeto e como este se integra no mundo atual.

##### 3.1.1 A evolução do setor de seguros de Londres

As origens do setor de seguros de Londres como o conhecemos hoje estendem-se por vários séculos, mas foi a partir do século XVII com a fundação de Lloyd's e o grande fogo de Londres que se começou a formar o mercado de seguros como ainda hoje opera.

Antes deste século, um momento crítico na evolução do setor foi o caso de Gybbons v. Martin em 1537, onde o tribunal decidiu a favor do segurado em casos de ambiguidade do contrato. Aparentemente alguns *underwriters* ou seguradores tinham concordado em fazer um seguro de vida para o Mr. Gybbons por um ano, e, com certeza, ele morreu uns dias antes do ano acabar. Mas os seguradores disseram que na verdade referiam-se a um ano lunar no contrato, que tinha acabado uns dias antes, claro que a Mme. Gybbons não aceitou e foi a tribunal que acabou por decidir em favor dela.

O caso estabeleceu o princípio de que qualquer incerteza na apólice seria resolvida a favor do segurado se fosse o segurador que redigisse o contrato. Esta decisão refletiu-se na indústria, causando uma solidificação do conhecimento do funcionamento e da regulação envolvendo estas matérias e contribuiu para a maturação do setor, sendo considerado um dos pilares na evolução sua evolução.

Em 1601, dois anos antes da morte da grande Rainha, um Ato do Parlamento visionava estabelecer um tribunal para “ouvir e decidir causas decorrentes de apólices de seguro”. Embora o tribunal não fosse muito bem-sucedido, demonstrou que os princípios do seguro eram conhecidos entre aqueles que com eles lidavam.

Edward Lloyd, nascido por volta de 1648, uma figura não diretamente envolvida em seguros, mas dono de uma cafetaria, desempenhou um papel fundamental no desenvolvimento do setor. Disponibilizou cabines na sua cafetaria para empresários dispostos a acartar uma parte do risco mediante uma taxa, e foi onde a tradição de ‘writing under’ começou. Os seguradores assinavam

nas apólices debaixo do risco, surgindo assim o termo “*Underwriter*”.

Em 1666, houve o Grande Incêndio de Londres, que devastou grande parte da cidade e consumindo inúmeras construções históricas, impulsionando mudanças no setor de seguros como a criação da primeira companhia de seguros de incêndio, The Sun. Isto incitou a mais movimento na cafetaria de Lloyd cujo negócio continuou a criar as bases para seguros marítimos e pagando quando necessário da parte dos seguradores como foi o caso, por exemplo, do naufrágio do Titanic[22]. A consistência de Lloyd’s em pagar as perdas permanece consistente há mais de três séculos, sendo uma referência sólida na indústria de seguros[23].

### 3.1.2 O setor de seguros de Londres contemporâneo

Em 2013, o setor de seguros representou impressionantes 20% do Produto Interno Bruto (PIB) da cidade e 8% do PIB de Londres. Esta significativa contribuição demonstra a importância económica do setor de seguros no contexto da capital britânica.

Adicionalmente, o mercado londrino destaca-se ao segurar ativos valiosos e únicos. Exemplos notáveis incluem a cobertura para as pernas do renomeado jogador de futebol Cristiano Ronaldo e a voz única de Bruce Springsteen. Esta capacidade de segurar riscos especializados sublinha a flexibilidade e abrangência do mercado de seguros de Londres, atendendo às necessidades específicas de indivíduos e personalidades públicas.

Como visto na Figura 6, o mercado de seguros de Londres assume a primeira posição global, sendo o maior centro mundial de riscos comerciais e especializados. Em 2013, o mercado londrino controlou mais de £60 bilhões em “GWP”, refere-se à quantidade de dinheiro paga à seguradora, fazendo deste o maior mercado mundial neste setor, superando significativamente a concorrência, com o segundo maior, o mercado de Bermuda, apresentando um tamanho de mercado de £25 bilhões em “GWP”[24], [25].

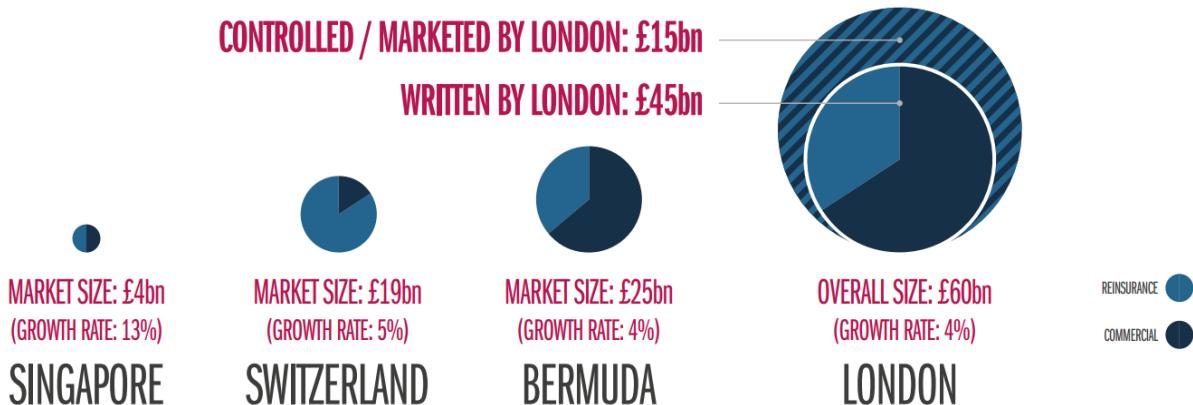


Figura 6: Maiores mercados de seguros - 2013

Fonte: [24]

### 3.1.3 Funcionamento e Terminologia de seguradoras

O mercado de Londres é um campo onde jogam muitas empresas ou sindicatos que serão representadas pelos *underwriters*. É normal subscreverem vários *underwriters* para dividirem o risco entre si.

O processo de emissão de seguros no século XVII mantém-se no século XXI: Aqueles que procuram segurar-se dirigem-se a um *broker* e explicam o seu risco de forma detalhada, a qualidade e preço das ofertas depende da quantidade das informações fornecidas. O *broker*, de seguida, vai ao encontro de *underwriters* que representam um sindicato de um investidor, procurando os que podem oferecer melhores taxas consoante o risco em causa. O *underwriter* pode dizer, por exemplo, “Aceito 8% a 2 por 250”, nesta situação este *underwriter* cobre 8% do risco, a uma taxa de 2 por 250 unidades monetárias. Desdobrando a proposta em mais detalhe:

- Uma taxa de 2 por 250 indica que se o segurado tem 250€ de propriedade segurada exposta a risco, o segurado teria que pagar um prémio de 2€, ou seja, teria que pagar pelo seguro 2€ anualmente (ou mensalmente conforme estipulado). Mas isto apenas se estiver segurado a 100%;
- Aceita 8% do risco, ou seja, cobre apenas 8% do valor caso o seguro se ative e recebe apenas 8% do prémio.

De seguida o *broker* dirigia-se a outros *underwriters*, já com a taxa estipulada e tenta segurar o máximo de risco possível. Por vezes 100% do risco é subscrito mas geralmente menos é segurado, e o proprietário ou comerciante é co-segurado para o resto, ou seja, é responsável pelo pagamento do restante[23], [26].

É frequente para projetos de alto capital serem segurados por bastantes *underwriters* dividindo o risco entre eles, segundo os documentos de seguros do Titanic, por exemplo, o navio encontrava-se segurado por mais de 70 *underwriters* diferentes[27], [28], como visto na Figura 7.

Figura 7: Titanic Placement Slip — 2013

Fonte: [28]

Com o processo geral estipulado pode-se agora detalhar as semânticas dos termos usados:

- **Underwriter:** O *underwriter*, ou subscritor, é o profissional ou entidade responsável por assumir o risco do seguro. Avalia o risco, discute e estabelece os termos da cobertura e define a taxa e o prémio;
  - **Broker:** O *broker*, ou corretor, atua como intermediário entre o segurado e os *underwriters*. Ele ajuda o segurado a compreender as suas necessidades, negocia com os *underwriters* em nome dele e facilita o processo de subscrição;
  - **Syndicate:** Um *syndicate*, ou sindicato, refere-se a um grupo de *underwriters* que se unem para assumir um determinado risco. Cada membro do sindicato contribui com uma parte da subscrição;
  - **Placement Slip:** É um documento preliminar que resume os detalhes essenciais de um contrato de seguro. Inclui informações sobre o risco, cobertura, termos e *underwriters*

envolvidos;

- **Endorsements:** Referem-se a alterações feitas a uma apólice de seguro existente. Podem incluir a adição, remoção ou modificação de coberturas, condições ou termos do contrato. Estes são utilizados regularmente para ajustar a apólice conforme as necessidades do segurado;
- **Sign & Close:** A etapa de “Sign and Close” refere-se ao processo de finalização de um contrato de seguro. Envolve a assinatura formal do contrato pelos participantes relevantes, confirmando a aceitação dos termos acordados. Este passo conclui o processo de subscrição e torna o contrato efetivo.

Este processo é comparável ao investimento numa empresa, *underwriters* podem envolver-se com empresas de banca ou de seguros com títulos de valor variável de empresas (“*securities*”). Da mesma forma que se pode lucrar ao investir num setor ou empresa, os *underwriters* acartam risco para gerar lucro[29].

## 3.2 Desenvolvimento *Low-Code*

Esta secção fornece uma resposta às questões de pesquisa sobre a definição do desenvolvimento *low-code*, o seu ciclo de vida, bem como características e especificidades relacionadas.

### 3.2.1 Definição de Desenvolvimento *Low-Code*

O termo “*low-code*” surgiu em 2014 num artigo da *Forrester Research* intitulado “New Development Platforms Emerge For Customer-Facing Applications” de Richardson C. [30]–[33], e desde a altura que tem ganhado tração e se solidificado como um campo da informática, num artigo de Alamin, [34] o desenvolvimento *low-code* é definido como um paradigma que prioriza o mínimo de programação no processo, isto através de programação visual com interface gráfica e *design* orientado por modelos. Noutras definições é também definido como um processo que envolve um esforço de programação mínimo com funcionalidades como *drag-and-drop*<sup>7</sup> e desenvolvimento acessível a programadores não profissionais[35].

### 3.2.2 Benefícios de Desenvolvimento *Low-Code*

O desenvolvimento *low-code* oferece diversas vantagens estratégicas, conforme destacado na lista abaixo. Ao acelerar o ciclo de desenvolvimento e envolver programadores menos especializados, esta abordagem responde eficientemente à escassez de profissionais especializados, permitindo

---

<sup>7</sup>Um paradigma de interface gráfica onde o utilizador pode selecionar elementos digitais, como ícones ou arquivos, movendo-os de um local para outro simplesmente ao clicar, manter o botão pressionado e soltando no destino desejado.

uma entrega mais rápida de aplicações. A redução de custos é alcançada através da eficácia no desenvolvimento, otimização de recursos e diminuição dos custos de manutenção. Esta prática também aumenta a capacidade de resposta às exigências do mercado e dos negócios, adaptando-se rapidamente a situações dinâmicas e cumprindo prazos apertados, minimizando o esforço de manutenção e promovendo uma colaboração mais eficaz entre as equipas de desenvolvimento e as empresas, fomentando um ambiente propício à inovação digital[35], [36].

- **Aceleração do ciclo de desenvolvimento:** Aproveita as funcionalidades da plataforma, minimizando a programação manual e incorporando um desenvolvimento visual de aplicações;
- **Envolvimento de programadores cidadãos:** Aborda a escassez de programadores especializados, contribuindo para uma entrega mais rápida de aplicações;
- **Redução de custos:** Alcançada através da rapidez do desenvolvimento, utilização eficiente de recursos e diminuição dos custos de manutenção;
- **Aumento da capacidade de resposta às exigências do mercado e dos negócios:** Adapta-se rapidamente a situações dinâmicas e cumpre prazos apertados;
- **Diminuição do esforço de manutenção:** Minimiza erros e problemas relacionados à integração, permitindo a contínua manutenção menos exigente;
- **Melhor colaboração entre equipas de desenvolvimento e empresas:** Facilitada por modelos visuais e colaboração frequente ao longo do ciclo de desenvolvimento;
- **Promoção da inovação digital:** Fomenta uma cultura de aprendizagem e experimentação;
- **Mitigação dos riscos de shadow IT<sup>8</sup>:** Através de ferramentas administradas pela organização a profissionais não pertencentes à área de TI, reduzindo soluções não autorizadas de TI e os riscos associados[35].

Em 2019 a OutSystems analisou o estado da área de desenvolvimento *low-code*, revelando as principais razões que impulsionam empresas a adotar estas metodologias como é possível observar na Figura 8, estas incluem: a aceleração da inovação e transformação digital, o aumento da capacidade de resposta às exigências do negócio, a redução da dependência de habilidades técnicas de difícil contratação, a mitigação de despesas devido a tecnologias obsoletas, a proteção contra a rápida evolução tecnológica, e permitir programadores cidadãos aprimorarem processos internos.

---

<sup>8</sup>Programas e sistemas de TI utilizados sem a aprovação do departamento de TI da organização, apresentando, portanto, riscos de segurança, integridade dos dados e outros.

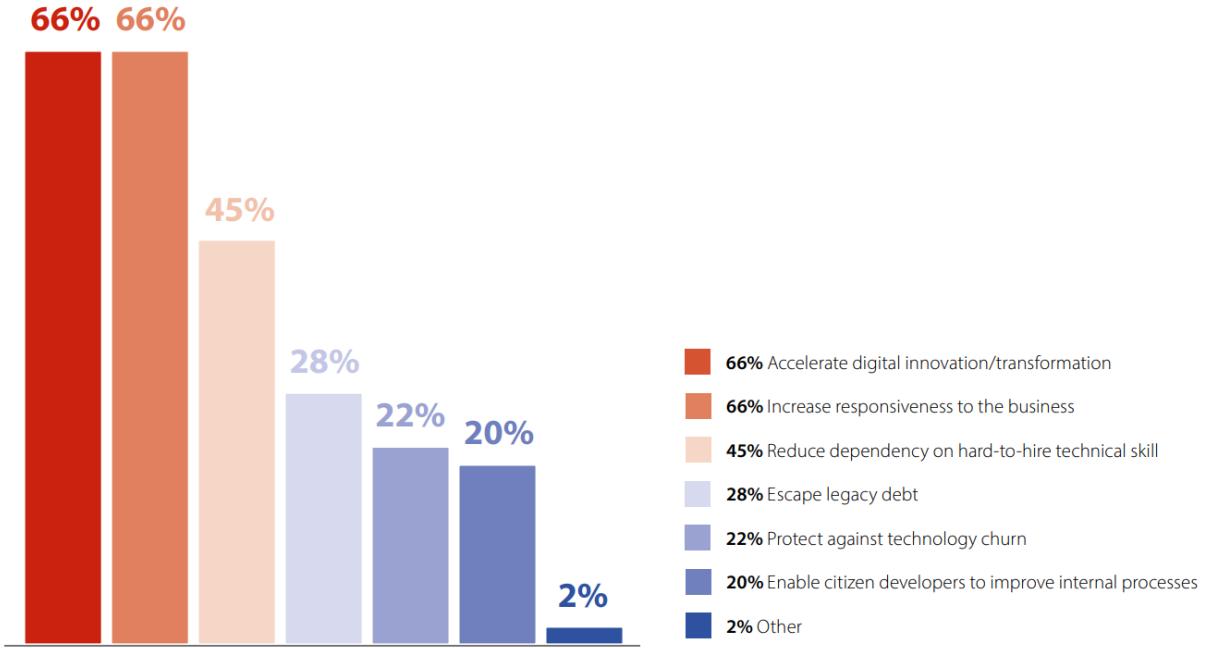


Figura 8: Maiores razões para usar *low-code*

Fonte: [37]

### 3.2.3 Limitações de Desenvolvimento *Low-Code*

A OutSystems identificou no seu estudo “The State of Application Development” de 2019, segundo os participantes os três desafios que mais dificultam a entrega de aplicações móveis e de web são:

- Integração com sistemas obsoletos ou ultrapassados;
- Requerimentos não bem definidos e inconstantes;
- Tempo necessário para testar.

Identificando também as maiores causas de retardam a entrega destas aplicações, como visto na Figura 9, sendo estas, a Integração de sistemas legados/APIs ausentes ou que necessitam de melhorias, Requisitos difusos/em constante mudança, Testes/Garantia de Qualidade, Proteção e teste de penetração em segurança, Preocupações com privacidade de dados, Falta de habilidades técnicas de desenvolvimento, *Design UX/UI* (incluindo *design* responsivo), Implantação/coordenação com operações de TI[37].

Segundo o artigo “The Impacts of Low/No-Code Development on Digital Transformation and Software Development” do autor Zhaochang Yan podem ser identificadas estas limitações no método:

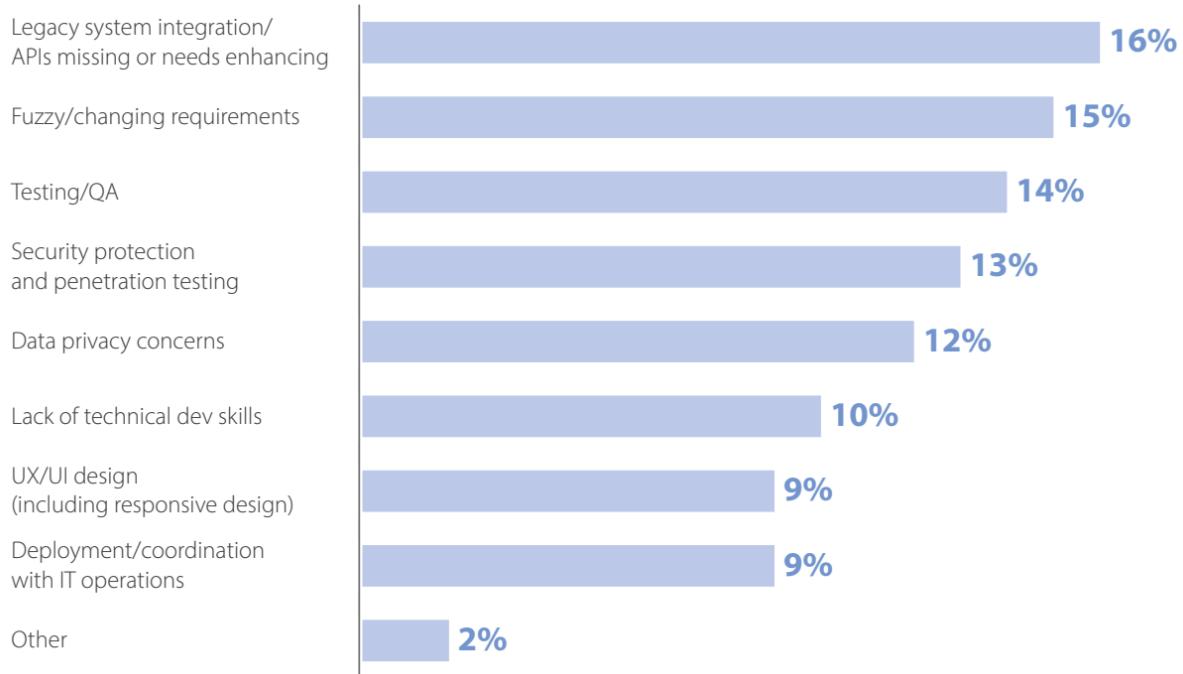


Figura 9: Maiores causas de atrasos na entrega de aplicações *low-code*

Fonte: [37]

- 1. Limitação na Customização/Flexibilidade:** Estas plataformas têm pré-implementações fixas em muitas áreas e casos, tornando-as menos customizáveis sacrificando flexibilidade. Programar funcionalidades customizáveis específicas é uma opção, mas pode ser difícil e moroso, podendo não ser o método ideal para aplicações que precisem de ser altamente customizáveis com um nível de complexidade mais elevado.
- 2. Limitação na Scalabilidade:** Muitas destas plataformas focam num processo célere na produção e entrega de aplicações de pequena a média escala, sendo uma ferramenta menos testada para aplicações de grande escala.
- 3. Preocupações de Segurança:** A utilização de plataformas *low-code* implica a inteira dependência da segurança da plataforma em uso. Organizações dependentes destas plataformas enfrentam riscos, uma vez que a segurança dos dados não está totalmente sob o seu controle. Se, por exemplo, os fornecedores da plataforma encerrarem as operações, as atualizações de segurança contínuas tornam-se incertas, deixando as organizações incapazes de lidar com novas vulnerabilidades.
- 4. Vendor lock-in:** Refere-se à dependência de um cliente em relação a um fornecedor, e à sua capacidade de mudar para outro. No contexto das plataformas *low-code*, à medida que investem mais num fornecedor específico, o processo de mudança torna-se mais

dispendioso e complexo e difícil[36].

Em suma, existem vários detalhes específicos ao desenvolvimento *low-code*, sendo importante analisar as vantagens expostas e as desvantagens identificadas e os seus pesos na implementação num dado projeto.

### 3.3 Terminologia do Projeto

Existem vários conceitos que serão frequentemente utilizados ao longo deste relatório e são frequentemente usados no contexto de trabalho, esta secção servirá para estabelecer bem os seguintes conceitos:

- **User Stories:** As *User Stories* (Histórias de Utilizador) são descrições detalhadas de funcionalidades do sistema, frequentemente utilizadas para comunicação entre os interessados e para orientar o desenvolvimento de software. Elas são a base para o desenvolvimento na plataforma;
- **Defects:** Uma incongruência identificada entre o funcionamento do programa e as USs (User Stories) definidas. São identificados com uma prioridade e registadas na Jira como descrito na Secção [6.6.1](#);
- **Incidentes:** Problemas submetidos por utilizadores da aplicação através da plataforma ServiceNow [\[6.2\]](#), o utilizador pode marcar a submissão de duas formas:
  - **Request:** O utilizador coloca esta *label* quando não tem a certeza se o problema está na forma de utilização ou de facto na plataforma;
  - **Incidente:** Se o utilizador está confiante que o problema com que se depara é de facto um problema da plataforma, marca desta forma.
- **GAP:** Caso específico em que a implementação está de acordo com a *user story*, mas a esta não contempla uma situação que se precisava de ter em conta.

Por exemplo: Se a *user story* for “Efetuar um programa que conte de um a cinco”, e o programa conte da seguinte forma: “1, 2, 3, 4, 5”, mas se descubra através de testes ou interações com utilizadores e confirmando com um BA ou alguém mais acima na hierarquia que esteja familiarizado com o que a aplicação tem que fazer, que a aplicação teria que contar também decimais desta forma: “1.1, 1.2, 1.3...”, então, como não estava bem explícito na US, considera-se um GAP.

É por isso que antes de reportar um novo *defect*, é geralmente necessário confirmar que o comportamento que acontece não está contemplado numa *user story*, porque se estiver, será um GAP. Neste caso a *user story* terá que ser revista, e a empresa cliente terá possivelmente

que pagar por outra US, visto que o pagamento se faz por USs;

- **Stamps:** Num contexto de lógica de seguradoras, os Stamps representam a aprovação formal de um *underwriter* para assumir um risco específico. Cada *underwriters* coloca seu carimbo no *Placement Slip* como uma indicação da sua subscrição no risco.

No contexto da plataforma, stamps são carimbos associados a uma organização cujos UWs pertencentes podem usar. Após a definição dos detalhes de um contrato e antes do envio da submissão aos UWs pertencentes, são atribuídos *roles*, estes *roles* são escolhidos consoante o stamp de cada UW. Podendo um UW não ter nenhum *role*;

- **Mutual Responsibility Contract(MRC):** O MRC, é um documento obrigatório a todos os contratos na plataforma que contém detalhes sobre o acordo que possam não estar contemplados na plataforma como informações de cobertura de uma apólice de seguro ou detalhes sobre o risco coberto, ou as condições e termos do contrato;
- **Endorsements:** Como explicado na Secção 3.1.3, referem-se a alterações feitas a um contrato, na plataforma existem os seguintes tipos de endorsement:
  1. Annual Resigning Statement;
  2. Cancellation;
  3. Decrease Cover;
  4. Increase Cover;
  5. Mid-Term Broker Change;
  6. Other Change to Cover;
  7. Reinstatement Provision;
  8. Tax Schedule;
  9. Mid Term Market Change (o único *endorsement* que requer um *Sign & Close*).

A termos de lógica de negócio a única diferença está entre o Mid Term Market Change e os restantes, devido ao seu requisito de ser feito num contrato *Signed and Closed*;

- **Corrections:** Correções são modificações realizadas para retificar erros ou informações incorretas em documentos relacionados ao seguro. Estas podem incluir ajustes nas apólices, documentos contratuais ou registos relacionados à subscrição. Na plataforma existem três tipos de *corrections*:

- **Cancel Contract:** A ação “Cancel Contract” permite ao *broker* cancelar completamente um *placement* de forma irreversível. A aceitação por parte da seguradora de um pedido de cancelamento reduz a linha escrita para zero e altera o estado do contrato para Cancelado. Uma vez concluído o cancelamento, é possível reutilizar o UMR.

No caso de cancelamento de uma *master facility*, as partes não acordadas serão canceladas assim que o líder da *facility* aceitar o cancelamento. Um pedido de cancelamento para qualquer *declaration* já vinculada que utilize a *facility* será acionado simultaneamente;

- **Cancel and Replace:** A ação “Cancel and Replace” está disponível ao longo do processo de *Firm Order*. Permite ao *broker* substituir um contrato cancelando o original simultaneamente de forma irreversível. A aceitação por parte da seguradora de uma resubmissão vincula a linha escrita ao novo contrato, reduzindo a linha no contrato anterior para zero e alterando o estado para Cancelado. Não é possível assinar e fechar até que todas as submissões originais sejam canceladas e todas as resubmissões sejam aceites;
- **Panel Change:** A ação “Panel Change” está disponível ao longo do processo de *Firm Order*. Permite ao *broker* fazer alterações nos detalhes da linha dos participantes no contrato. Incluindo em painéis de *facilities*, desde que o líder da *facility* não seja um Slip, Lloyds, XIS ou LIC Lead no *placement*. Detalhes da linha que podem ser alterados incluem linha escrita e stamp. A ação é irreversível, sendo necessário responder a todos os pedidos de “Panel Change” antes de assinar e fechar.
- **Master Facilities:** Master Facilities ou Facilities referem-se a um conjunto de seguradoras que se juntaram e aceitaram fazer parte da *facility*, esta que depois pode ser adicionada como painel conjunto a um contrato;
- **Declarations e Open Market Contracts:** *Declarations* e contratos de mercado aberto referem-se a contratos que utilizam painéis de *facilities* ou não. As *declarations* são contratos que envolvem uma ou mais *facilities*, já contratos de mercado aberto envolvem apenas UW individuais;
- **Off-platform Underwriters:** *Underwriters* fora da plataforma são profissionais que assumem riscos de seguro, mas que não utilizam diretamente a plataforma para a subscrição. A interação eficaz com esses *underwriters* é simulada na plataforma por um UW *off-platform* cujas respostas ao contrato são controladas pelo *broker*, desta forma este deve alinhar com estes UWs fora da plataforma e consoante o decidido, colocar a resposta por eles na plataforma;

- **Archive Placements:** Refere-se ao armazenamento organizado de regtos e documentos relacionados a colocações de seguros já fechados. É muito usado na transferência de dados de uma plataforma antiga;
- **Placements:** Referem-se a uma estrutura que pode conter vários contratos. É obrigatório um contrato se encontrar sempre dentro de um *placement*;
- **Firm Order:** É o processo vinculativo executado pelo *broker* que manda os pedidos de aceitação do cancelamento ou do endorsement ou das linhas do contrato para os UWs. Pelo que mandar uma *firm order* será mandar o pedido em contexto para os UWs apropriados;
- **Retrofit:** É a passagem de código de um ambiente mais avançado, ou seja, mais próximo do ambiente de produção, para um ambiente de mais baixo nível, como por exemplo um dos ambientes de desenvolvimento, isto pode ser necessário, por exemplo, se um *defect* teve que ser resolvido num ambiente mais alto de forma a que a sua resolução chegue mais rapidamente a produção, neste caso ter-se-ia que trazer este código para os ambientes mais baixos também, a este processo chama-se retrofit, refira à Secção 5.3 para mais informação sobre ambientes;
- **Cancel Line:** Cancelar uma linha representa a anulação de uma linha específica num contrato de seguro, ou seja de um UW ou de uma *facility*. Após os relevantes terem aceite, se não tiverem nenhum *role* gerado pelos seus stamps, o *broker* pode lhes fazer um cancel line, que irá mandar um pedido para se retirarem;
- **Withdraw:** A ação de withdraw está disponível depois de fazer uma submissão para um UW, mas antes deste aceitar. Ao fazer *withdraw*, o UW não poderá mais aceitar o pedido;
- **Business Analysts (BA):** Os *Business Analysts* analisam e compreendem os processos e funcionamento de negócios, identificando oportunidades de melhoria, desempenhando tarefas como análise de dados, comunicação com clientes, documentação de processos, testes e validações.

Na prática, os BA's são generalistas da equipa que têm uma boa noção de como a aplicação se deve comportar e a quem se recorre em caso de dúvida.

## 4 Organização do Projeto e Empresa

Nesta secção será explorada a estrutura e organização da Deloitte e do projeto no que toca a recursos humanos, licenças, programas, e a integração com os restantes mecanismos de gestão da Deloitte PT. Devido à grande escala do projeto RIL este envolve uma colaboração extensa entre a Deloitte PT, a Deloitte UK e a Deloitte India.

### 4.1 Estrutura Organizacional da Deloitte PT

A Deloitte Portugal organiza-se de forma a proporcionar uma integração eficiente dos novos contratados, facilitando o seu desenvolvimento de carreira e apoio contínuo aos colaboradores. Nesta secção, abordaremos as principais práticas e elementos relacionados à estrutura organizacional da Deloitte PT.

#### 4.1.1 Formações e Integração

As formações, como já falado no primeiro capítulo, desempenham um papel crucial na integração de novos colaboradores à equipa em que são inseridos enquanto elevam simultaneamente as capacidades dos novos recrutados a termos de ferramentas que são frequentemente usadas na empresa ou em projetos da empresa. Servem também como uma avaliação inicial das aptidões de cada “new hire” para mais facilmente identificar um projeto que se adeque a ele.

#### 4.1.2 Hierarquia e Progressão de Carreira

Cada profissional tem um estatuto para com a hierarquia da empresa. Esta hierarquia dita quem tem poder de decisão sobre quem, quem é que faz os “check-ins”, e reflete-se também no salário de cada um.

**Check-Ins:** Os “check-ins” são reuniões mensais realizadas entre cada membro da Deloitte e o seu superior. Durante estas reuniões, são avaliados pontos fortes e fracos desde o último encontro, identificam-se áreas de melhoria e é dada uma nota. Estes encontros são essenciais para manter o desenvolvimento profissional positivo e saber quais são os pontos de melhoria dos profissionais, permitindo um contínuo aprimoramento das suas habilidades. São também estabelecidas metas, objetivos anuais ou mensais.

Existem vários estatutos hierárquicos, por ordem crescente de hierarquia, os seguintes:

- **Trainee**

Um estatuto inicial na Deloitte, reservado para aqueles que estão nos primeiros etapas de suas carreiras. Os trainees geralmente estão a ter a sua primeira experiência de trabalho na área ou estão a trabalhar no contexto de um estágio curricular ou profissional e estão em

processo de aprendizagem e desenvolvimento;

- **Programmer**

A designação para quem foi contratado como programador e tenha pouca experiência, e um a dois anos;

- **Experienced Programmer**

Após algum tempo e feedback positivo, os Programmers são promovidos ao estatuto de Experienced Programmers;

- **Team Leader**

O estatuto de Team Leader é alcançado por indivíduos que estão na empresa há algum tempo e demonstraram já capacidade de liderança e tenham uma boa ideia dos funcionários internos da empresa. Nesta posição, assumem a responsabilidade por uma equipa, estando encarregados de realizar os Check-ins com os restantes membros.

Podem também, a partir deste título, assumir o papel de “Career Coach” para outros membros da empresa;

- **Pro Manager**

Enquanto os papéis anteriores necessitavam ainda de um grande contacto com o projeto e as tarefas em si, os seguintes são papéis que envolvem maioritariamente gestão de recursos.

Têm um papel-chave na gestão de equipas e projetos;

- **Manager**

Uma posição de gestão mais avançada, envolvendo mais responsabilidades significativas na coordenação e supervisão de projetos;

- **Senior Manager**

O último nível a que se pode ser promovido dentro da categoria de gestão com mais poder e responsabilidade na liderança estratégica dos projetos e gestão do cliente;

- **Associate Partner**

Uma posição associada já à gestão de colaborações da Deloitte, começam a ter responsabilidades em relação aos novos contratados e a que projetos é que são alocados, bem como o contacto e fecho de contratos com empresas parceiras;

- **Partner**

O estatuto mais elevado, com acrescidas responsabilidades de gestão de projetos e de empresas parceiras.

#### **4.1.3 Buddys**

Buddys são um sistema da Deloitte para mais facilmente integrar novos contratados. Normalmente entram em contacto no início da contratação e servem como alguém com quem o indivíduo se possa sentir confortável a fazer perguntas gerais acerca da empresa, do funcionamento da empresa, da equipa, etc. Estão num nível hierárquico idêntico a quem são atribuídos, mas, estão na empresa à mais tempo.

São uma forma de fazer um recém-contratado se sentir mais bem integrado num ambiente novo.

#### **4.1.4 Career Coaches**

A cada novo elemento na empresa é também atribuído um “Career Coach”, alguém com quem se possa falar sobre o ambiente na sua equipa ou como se sente em relação ao trabalho sem ser diretamente da mesma equipa. Serve também para aconselhar acerca da progressão de carreira na Deloitte, é normalmente marcada uma reunião mensal com o “Carrear Coach” atribuído.

#### **4.1.5 Trabalhadores externos**

O funcionamento dos trabalhadores externos na Deloitte segue um modelo de subcontratação, conhecido como *outsourcing*. Enquanto os funcionários internos são contratados diretamente pela Deloitte, os trabalhadores externos são alocados aos projetos através de uma empresa contratada.

A Deloitte estabelece contratos com empresas externas para fornecer profissionais que serão alocados a projetos específicos. Neste modelo, a Deloitte realiza pagamentos à empresa terceira pelos serviços prestados, e esta última é responsável por fornecer os recursos humanos necessários.

Esta abordagem permite à Deloitte flexibilidade na gestão da sua frota de trabalho, garantindo a disponibilidade de especialistas conforme necessário para atender aos requisitos de projetos específicos. Ao mesmo tempo, as empresas externas fornecem profissionais especializados para atender às necessidades da Deloitte, criando uma colaboração eficaz no âmbito de *outsourcing*. Este é de forma geral também o modelo de trabalho da Deloitte para com outras empresas.

#### **4.1.6 Engagement e Timesheet**

A gestão de projetos na Deloitte PT é acompanhada de perto através de ferramentas como *Engagement* e *Timesheet*. A cada quinzena é obrigatório a todos os trabalhadores da Deloitte o preenchimento e a submissão da *timesheet* com informações das horas trabalhadas bem como os *engagements*, para ficar registado que tarefas ou projetos é que foram trabalhados.

O sistema de *Engagement* é fundamental para planear e oferecer uma visão geral da progressão de cada indivíduo, é também uma forma de comprovar aos parceiros que trabalho é feito bem como quantos trabalhadores e horas são debitadas em que projetos. É usado também para efeitos de faturação.

Estas ferramentas não apenas auxiliam na alocação eficiente de recursos, mas também fornecem conhecimentos valiosos sobre a distribuição do esforço das equipas ao longo dos projetos. Esta transparência contribui para uma melhor compreensão no investimento de tempo em diferentes áreas do trabalho e o compromisso com o uso diligente delas reflete a constante procura por eficiência e qualidade nos serviços prestados pela Deloitte.

## 4.2 Estrutura Organizacional da RIL na Deloitte PT

A estrutura do projeto da RIL dentro da Deloitte é fundamental para a garantir o seu sucesso e a eficiência, e foimeticulosamente delineada para maximizar os ganhos do trabalho individual no projeto como um todo.

### 4.2.1 Reuniões KT

Uma das ferramentas usadas nesta organização são, por exemplo, as reuniões KT:

**Reuniões KT:** As Reuniões KT são sessões que se têm em intervalos de tempo irregulares onde os trabalhadores do projeto se juntam todos e é discutida uma nova funcionalidade, de uma parte da aplicação ou uma implementação, onde são encorajadas também perguntas. Estas reuniões são gravadas e disponibilizadas para todos os envolvidos.

### 4.2.2 Formações Introdutórias

Existem algumas formações introdutórias que cada novo elemento no projeto tem que seguir, as formações a completar são estabelecidas consoante a equipa do recém-chegado. No caso da equipa AMS RUN, os novos membros têm que completar as seguintes formações:

- Formação de OWASP da Udemy através da Deloitte;
- [Formação introdutória a MongoDB](#);
- [Formação de aggregations de MongoDB](#);
- Uma lista de demo vídeos acerca do funcionamento da plataforma;
- Parte da documentação técnica do projeto.

#### 4.2.3 Prioridades e 24/7

Outra forma de organização é através da categorização de gravidade de incidentes e alertas. Neste contexto, são adotados diferentes níveis de prioridade, representados por P1, P2, P3 e P4 que determinam a urgência de cada incidente onde P1 é o mais crítico. Por vezes há alertas críticos mesmo fora de horas de trabalho, por isso foi necessário a criação de um papel, o 24/7, no qual um indivíduo fica elegível para lidar com incidentes graves a qualquer momento do dia caso apareçam. Esta posição rotativa requer disponibilidade contínua sendo remunerada adequadamente.

#### 4.2.4 Equipas

A termos de estrutura, dentro da Deloitte PT, a organização da RIL é dividida em grupos e equipas específicas, cada uma com responsabilidades distintas. Destacamos dois principais grupos:

##### 4.2.4.1 Grupo RUN

O Grupo RUN é responsável pela manutenção da aplicação e correção de incidentes. Este grupo está subdividido em várias equipas, cada uma focada numa área específica:

- **Infraestrutura (Azure):**

Equipas responsáveis pela gestão da infraestrutura na plataforma Azure, incluindo a manutenção dos serviços Microsoft integrados com a plataforma, como as contas usadas para testar e aceder à plataforma, e a sua autenticação, feitas através do Microsoft ActiveDirectory;

- **MongoDB:**

Equipas que gerem a base de dados de MongoDB e questões relacionadas ao desempenho e às pesquisas da base de dados;

- **OutSystems (OS):**

O ramo de OutSystems está também dividido em várias equipas na Deloitte PT, existem tarefas predefinidas que alternam regularmente entre estas equipas, normalmente de 3 em 3 semanas:

- **Triagem:**

Equipa focada em incidentes de produção, o *team leader* deverá atribuir uma prioridade aos incidentes e distribuí-los pela equipa;

- **Equipa Lights On:**

Equipa responsável pela resolução de *market defects*, ou seja, defeitos associados a

incidentes de produção. Estão também encarregues, simultaneamente, das tarefas semanais, estas são tarefas como:

- *Deploys*: Passagem de código de um ambiente para outro de acordo com o *Deployment Aggregator*;
- Scans de segurança;
- Scans e questões de desempenho;
- Scans de processos.

- **Resolve defects:**

Apenas resolve *defects* não associados a *defects* de produção, ou seja, qualquer defeito detetado em testes, durante o desenvolvimento ou durante a resolução de outros *defects*. São também responsáveis por preparar a nova *release*;

Estas equipas estão muito interligadas, e é frequente se misturarem tarefas quando necessário, por exemplo, numa época com alta procura em que haja muitos incidentes de produção é frequente haver alguns membros de algumas equipas a ajudarem na triagem. Para além destas três equipas rotativas, existe também uma quarta equipa temporária:

- **Surge Team:**

A equipa Surge tem objetivos específicos bem definidos. É uma equipa formada por vários elementos das restantes equipas com diferentes graus de conhecimentos em diferentes matérias permitindo-lhe abranger uma grande amplitude de problemas, o único objetivo desta equipa é resolver *defects*.

A equipa tem uma *release* própria e um conjunto de incidentes a resolver previamente acordados com o cliente, tem metas específicas e prazos de entrega definidos.

A criação desta equipa deu-se devido à necessidade constante de pessoal em triagem, o objetivo é a correção de defeitos ou fontes de bugs na aplicação que estão a fazer com que muitos incidentes sejam abertos na triagem, e ao mesmo tempo, baixar o número de incidentes no *backlog* do projeto.

#### 4.2.4.2 Grupo Delivery

A equipa Delivery é responsável pelo desenvolvimento, casos de uso e implementação de novas funcionalidades. Este grupo trabalha de forma estratificada, dividindo-se em PODs, equipas com objetivos específicos, que seguem a metodologia de *sprints* para o lançamento de *releases*. Cada *release*, por exemplo, 2.4, é composta por um conjunto de X *sprints*.

As PODs, cujas tarefas são rotativas, trabalham em *sprints*, focando-se em tarefas como *User Stories*. Antes de iniciar um *sprint*, são realizadas sessões de *Detail Design (DD sessions)* para escrever e rever as *User Stories*, garantindo clareza e tirando quaisquer dúvidas que possam existir.

No dia da entrega de uma *User Story* há uma reunião com a equipa de testes (BAs), onde a *User Story* é apresentada conforme o seu “happy path” ou o caminho principal, para garantir que esteja a funcionar conforme o esperado.

Algumas equipas específicas no Grupo Delivery incluem a equipa de APIs (foca-se na manutenção dos APIs de MongoDB e Azure) e a Data Team (trabalha com a base de dados MongoDB).



## 5 Infraestrutura Tecnológica do RIL

Nesta secção, vamos explorar a infraestrutura tecnológica essencial para o projeto, abrangendo a plataforma OutSystems, ferramentas de integração, nuvens e outros componentes vitais.

### 5.1 Integração com Ferramentas na Nuvem

No projeto temos uma forte integração com uma panóplia de ferramentas presentes na nuvem com uma forte e complexa rede de APIs que interligam os diferentes sistemas entre si de forma segura e eficaz.

#### 5.1.1 Azure

O Azure desempenha um papel fundamental no projeto e muitas funcionalidades desta nuvem são extensamente utilizadas, como por exemplo:

- **Azure AD**(Active Directory): Usado para autenticação dos utilizadores;
- **Máquinas Virtuais**(VMs): Várias máquinas virtuais são usadas por várias razões.

Temos duas regiões ativas para as VMs, UK South e UK West, para um determinado utilizador é selecionado aquele servidor relativamente ao qual o utilizador apresenta menor latência. Este procedimento serve como uma camada de segurança contra desastres naturais ou humanos caso um dos *datacenters* seja comprometido.

Existem dois ambientes principais no Azure do projeto, *RIL* e *RIL SandPit*, onde o primeiro contém todas as VMs relacionadas com o ambiente de produção e o segundo todas as relacionadas com os outros ambientes e as suas diferentes finalidades. Cada ambiente tem a sua própria máquina virtual atribuída, no entanto os conceitos de *scale up* e *scale out* são relevantes na sua organização;

**Scale Up:** Refere-se ao aumento da capacidade de um sistema ou máquina virtual aumentar os recursos de um ou mais nodos já existentes da rede, ou neste caso, de uma só VM. Isto pode envolver o aumento do poder computacional da máquina como aumentar a quantidade de CPU, memória RAM ou armazenamento. Geralmente, este método é usado quando é necessário fazer uma tarefa única que requer muitos recursos apenas da máquina que a vai realizar, como, por exemplo, um *deploy* bastante extenso;

**Scale Out:** Diz respeito à expansão horizontal do sistema, cuja metodologia se foca na distribuição da carga entre vários nós ou servidores em vez de fortalecer um único dispositivo. É mais eficaz para lidar com cargas de trabalho dinâmicas, e é útil, por exemplo, nas horas de maior ou menor afluência de utilizadores[38];

- **Azure Firewall:** Utilizado para o controlo de tráfego no ambiente Azure, gerir e garantir a segurança das comunicações;
- **Azure PagerDuty:** Para gerar alertas críticos e mais fácil e eficazmente responder a estes;
- **Azure Storage & logs:** O armazenamento não alocado às VMs tem a função de guardar logs de todas as partes da aplicação. São utilizadas no projeto em conjunto com, por exemplo, as **Logic Apps** para criar e orquestrar fluxos de trabalho automatizados como, por exemplo, inspecionar as tabelas de SQL de OutSystems dentro das VMs com a componente de Azure de SQL e retirar os logs relevantes e armazená-los. Também servem para inspecionar os logs em si, e caso aconteça algo fora do normal, enviar um alerta PX. São usadas também **Function Apps** desenvolvidas em .NET ou Python nos casos que pedem um pouco mais de controlo, como interações com os *endpoints* do MongoDB.

### 5.1.2 Atlas

O Atlas é a nuvem de armazenamento do MongoDB, é usado no caso do projeto para armazenar todos os dados relacionados com a base de dados, desempenhando um papel central na gestão e teste destes.

Muitas vezes é necessário fazer *datafixes*, uma alteração à base de dados para os dados ficarem como o cliente espera. Muitas vezes *datafixes* são necessários devido a um problema no código que causou alguns dados da BD a serem guardados numa forma indevida, ou alguma ação que o utilizador fez que tenha que ser revertida. *Datafixes* são também muitas vezes usados mesmo em ambientes de teste, facilitando o processo de reprodução de problemas.

Para mais informação acerca do uso da plataforma, por favor consulte à secção [Plataforma Atlas](#).

### 5.1.3 Nuxeo

O Nuxeo é outra peça-chave no projeto, integrando-se com o [Power Config API](#) é usado para armazenar os documentos carregados pelos utilizadores na aplicação.

É importante que esteja bem integrado com Azure para impedir que os URLs dos documentos não estejam acessíveis a qualquer um, pelo que tem que ter um registo dos utilizadores e das permissões de cada um. Serve como uma base de dados para guardar os documentos sensíveis dos utilizadores como os contratos, documentos de mudanças de contratos, certificados, relatórios, etc.

### 5.1.4 SendGrid

Embora não faça parte da infraestrutura de nuvem que manipula informações sensíveis, [SendGrid](#) é utilizado como uma solução externa para o envio de e-mails para os utilizadores. É um API que é usado exclusivamente pelo código em OutSystems.

## 5.2 APIs

Nesta secção vamos explorar como os APIs do projeto funcionam e estão expostos para que todas estas ferramentas distintas possam comunicar entre si com sucesso e de forma segura.

No início do projeto, uma grande parte do código encontrava-se em APIs externos em código .NET e C#, para, por exemplo, OutSystems ter acesso à base de dados de Mongo. No entanto com o tempo, a tendência foi converter estes blocos de código em lógica que pudesse correr mesmo dentro da OutSystems sob a forma de extensões ou módulos. No seu estado atual, o projeto tem os APIs internos de OutSystems e o API Org Config.

### 5.2.1 APIs OutSystems

Trata-se de APIs internos da OutSystems usados para comunicação entre módulos ou serviços do ecossistema de OutSystems como o Service Studio e o Service Center.

Dentro da próprio OutSystems também foram criados vários módulos *\_Drv* que são a segunda camada mais baixa da organização no projeto, estes módulos utilizam uma extensão da OutSystems que se pode visualizar através do Service Center como demonstrado na Figura 10, foi criada com o propósito exclusivo de comunicar com a base de dados de MongoDB que é um exemplo de uma das formas como o projeto evoluiu para ser integrado de uma forma mais estreita com o ecossistema de OutSystems.

### 5.2.2 API Org Config

Desenvolvido em .NET fora de OutSystems, o papel do Org Config é ter um método de comunicar com mongoDB, Nuxeo, Azure e OutSystems de forma consistente, garantindo não haver dessincronização entre os dados.

Por exemplo, o registo dos utilizadores que existem em cada ambiente e as suas permissões devem constar da base de dados MongoDB, o principal sítio que mantém os dados dos utilizadores, no Nuxeo, para que um documento de um cliente possa ser acedido apenas por esse cliente e o URL não seja público.

Estão guardados também no Azure, para efeitos de análise de dados e erros.

Desta forma o processo é simplificado, exportando a lógica de sincronização entre vários

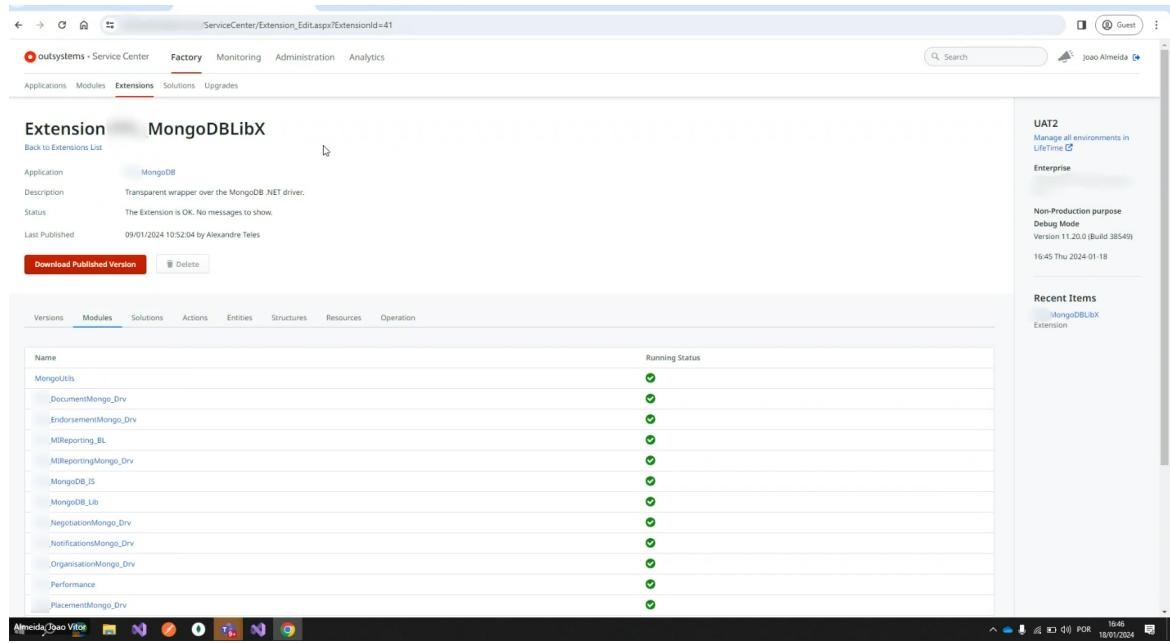


Figura 10: Extensão da OutSystems

Fonte: Documentação Interna

serviços para o API Org Config, que disponibiliza a lógica em mais de uma dúzia de funções visíveis na Figura 11.

### 5.3 Versionamento

Como alternativa à utilização de uma ferramenta direta de controlo de versionamento, foi elaborado um sistema de vários ambientes, cada um com uma função específica, por onde o código viaja antes de finalmente se encontrar no ambiente final de produção usado pelos utilizadores.

Os números das versões regem-se por majors, minors e hotfixes.

Version: 2 . 3 . 5 .  
 major      minor      hotfix

- **Major:** A versão principal que indica grandes atualizações ou alterações no software. Aumentar o número da versão principal implica mudanças significativas como, por exemplo, uma reestruturação da plataforma e pode resultar em incompatibilidades com versões anteriores;
- **Minor:** Reflete atualizações menores, melhorias ou adições em partes focadas da aplicação,

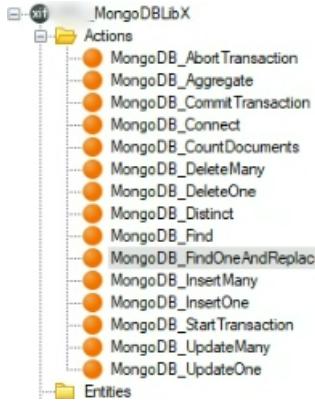


Figura 11: Usable actions by the API

Fonte: Documentação Interna

como, por exemplo, alteração do fluxo para criar um contrato;

- **Hotfix:** Uma correção rápida para resolver problemas críticos, bugs ou *defects*. A diferença entre hotfixes não costuma ser significativa, pelo que não é geralmente referida. Quando se menciona a versão da plataforma, geralmente inclui-se apenas da major e da minor, por exemplo: Versão 2.3.

### 5.3.1 Ambientes

Nesta seção, apresenta-se uma visão abrangente dos ambientes envolvidos no processo de desenvolvimento, teste e implementação.

É crucial compreender o propósito de cada ambiente para garantir uma implementação eficaz da metodologia AD Factory. A Tabela 2 oferece uma descrição detalhada dos diferentes ambientes, destacando a sua utilização principal e os principais utilizadores envolvidos.

**AD Factory:** A AD Factory (Application Development Factory) representa uma abordagem organizada no projeto, focada na estruturação eficiente do ciclo de desenvolvimento. Divide-se em várias linhas de trabalho (PODs) que se refletem em várias equipas, a AD Factory coordena desde a fase de design até aos testes num processo estruturado. Durante a fase de design, as decisões detalhadas são tomadas em cada POD, com histórias de utilizadores aprovadas por um Analista de Dados de Negócios (BDA). A fase de construção e testes seguem-se conforme planos elaborados em sessões de *sprint*, utilizando quadros JIRA específicos para cada POD. Acompanha-se o progresso através do JIRA, com relatórios, fluxos principais e a lista de *defects* bem categorizados.

Tabela 2: Descrição dos ambientes

Fonte: Documentação Interna

Ambiente	Descrição / Uso	Principais Utilizadores
Desenvolvimento (DES)	Ambiente de desenvolvimento para desenvolver as “User Stories”.	Programadores
Teste (TES1)	Utilizado para teste de sistema com DES, integração e regressão.	Programadores e Tests na sprint
Desenvolvimento para AD Factory (DED)	Ambiente para apoiar o desenvolvimento do ramo de código AD Factory em paralelo com as correções e correções de bugs no ramo DES.	Programadores da AD Factory
Teste para AD Factory (TES2)	Ambiente temporário para apoiar os testes do ramo DED.	Programadores da AD Factory e Testes na sprint
Ponta a Ponta (P2P)	Utilizado para testes ponta a ponta de temas discutidos nas <i>sprints</i> .	Equipa de Testes E2E
Aceitação do Utilizador (ADU)	Utilizado para testes de aceitação do utilizador. Teste de upload de configuração da organização e verificação de processos com empresas.	Equipa de Testes da RIL
Performance (PCE)	Utilizado para testes de desempenho.	Equipa de Testes de Performance
Pré-Produção (PREP)	Ambiente para auxiliar o ambiente PRO, para testes, configuração, testes de mercado e integração de APIs.	Equipa de Testes de Mercado da RIL
Teste Integração Conjunta (TIC)	Utilizado para integração do mercado com a API exposta externamente.	Devs e entidades externas
Demonstração (SHOW)	Ambiente com cópia de PRO para efeitos de demonstração.	Gestão de Relações RIL
Produção de Suporte (AUX)	Ambiente publicado com uma versão do código de produção quase idêntico, permitindo solucionar problemas sem impactar o ambiente de produção.	Equipa de Execução
Produção (PRO)	Ambiente de produção disponível aos clientes. Testes de Aceitação Operacional. Teste de Penetração da RIL.	Clientes da RIL, administradores da RIL e Equipa de Execução

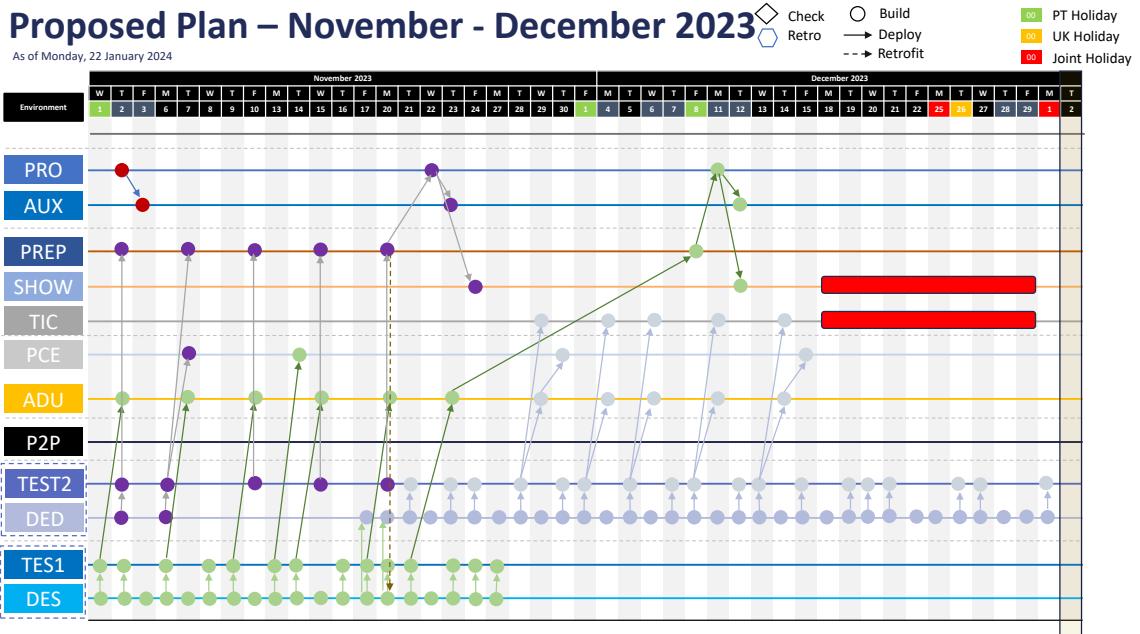


Figura 12: Deployment Aggregator

Fonte: Protótipo de um *Deployment Aggregator* com base na Documentação Interna

### 5.3.2 Deployment Aggregator

A cada par de meses, é gerado um novo “Deployment Aggregator”, um cronodiagrama que detalha o fluxo do código entre ambientes e as datas em que estes *deploys* devem ocorrer.

Pode ter também a representação de férias ou freezes, um paradigma bastante usado em empresas, que se refere a períodos em que não há textitdeploys e a maior parte dos trabalhadores não estão de serviço, por exemplo, por estarem de férias, podendo, no entanto, haver *deploys* de emergência. Este período é representado por retângulos vermelhos no *deployment aggregator*, Figura 12.

Existem dois ambientes de desenvolvimento, neste exemplo, TEST1 e TEST2, ambos com um ambiente de testes, DES e DED, respectivamente. Foram criados os dois para possibilitar o trabalho em duas *sprints* de desenvolvimento, duas minors diferentes, por exemplo, simultaneamente. No protótipo de exemplo dado, está-se a trabalhar na minor 2.4 no ambiente DES e na minor 2.5 no ambiente DED. É possível ver que o código da 2.4 vai chegar a produção no dia 11 de dezembro, e que o código com a 2.5, que irá ser fundido no ambiente ADU, não tem ainda uma data definida para a chegada à produção devido ao freeze.



## 6 Ferramentas e Plataformas usadas no desenvolvimento da RIL

Nesta secção vão ser analisadas as ferramentas usadas no projeto, e as equipas ou profissionais que normalmente as manuseiam e como.

### 6.1 OutSystems

Estabelecida em 2001, a OutSystems é uma das principais plataformas mencionadas no desenvolvimento *low-code*, foi desenhada com o fluxo empresarial em mente, facilitando a criação e entrega rápida de aplicações. A sua interface de desenvolvimento visual permite a construção de aplicações usando modelos visuais, ações, módulos e blocos reutilizáveis, e diferencia-se oferecendo um conjunto abrangente de ferramentas que cobrem todo o ciclo de vida da aplicação[39]. De seguida são descritos os diferentes ambientes do ecossistema de OutSystems que se usam de forma regular no ambiente de trabalho.

#### 6.1.1 Service Studio

O Service Studio é o ambiente de desenvolvimento low-code de OutSystems, é onde se criam as aplicações, módulos e a lógica por detrás destas, bem como o desenvolvimento das interfaces de utilizadores para aplicações, quer sejam web tradicionais, web reativas ou móveis. É neste ambiente também que se definem os modelos das bases de dados internas a OS.

É uma ferramenta muito importante no contexto diário para o desenvolvimento e manutenção cotínuas da aplicação.

#### 6.1.2 Service Center

A plataforma oferece uma variedade de configurações que podem ser ajustadas para personalizar o comportamento em diferentes áreas. Estas configurações são ajustáveis a partir do Service Center, permitindo a visualização e gestão de uma panóplia de funcionalidades como módulos e dependências entre eles, gestão das diferentes aplicações do projeto, gestão do versionamento de cada módulo através da publicação da versão desejada. Bem como o ajuste de variáveis globais.

O Service Center é também uma ferramenta essencial para gerir e diagnosticar erros durante o desenvolvimento ou manutenção, permitindo visualizar logs relacionados com timers, processos, módulos, *screen requests*, *service actions*, *traditional web requests*, integrações, extensões ou e-mails.

### 6.1.3 Lifetime

O Lifetime é uma plataforma do ecossistema de OutSystems que gera o ciclo de vida completo das aplicações. Proporciona visibilidade total sobre os diferentes ambientes, desde o desenvolvimento até à produção, sendo a ferramenta principal na realização de *Deployments*, resolvendo conflitos e efetuando *merges* automaticamente sempre que possível e ajudando o utilizador no processo em caso de conflitos. Oferece, além disso, funcionalidades para versionamento e migração.

### 6.1.4 Um Guia de Boas Práticas de Código para OutSystems

Todas as linguagens de programação bem estabelecidas têm diretrizes de programação tão bem enraizadas na cultura da língua que são ensinadas em conjunto com ela, desta forma, com os profissionais a aplicar um método parecido no desenvolvimento do código tem-se um projeto muito mais coerente.

No entanto, OutSystems carece de um guia de diretrizes estabelecido. Isto deve-se principalmente ao facto de OutSystems ser radicalmente diferente de outras linguagens de programação, sendo um paradigma visual onde até a interface pode ser modificada, não existe um consenso universal para a forma como o código deva ser organizado visualmente.

Neste capítulo vamos explorar algumas das diretrizes que nos foram introduzidas e aconselhadas a seguir quando alterássemos código na aplicação. Estas indicações começaram a ser introduzidas recentemente, pelo que muito do código existente não cumpre estas normas, mas há um incentivo para uniformizar o código da aplicação.

**Diretriz 1 — Sentido descendente é progresso:** Figura 13, o sentido descendente deve levar-nos ao objetivo da ação. Este é o sentido de leitura geralmente acordada, até por diferentes culturas;

**Diretriz 2 — Ramificações para a direita:** Figura 14, o ramo mais importante de um “if” deverá seguir da esquerda para a direita. Um nodo “if” geralmente tem um ramo que se espera ser executado com mais frequência ou o ramo que é simplesmente mais importante a termos de valor comercial. Este ramo deve mover-se para a direita, e o outro deve continuar reto para baixo;

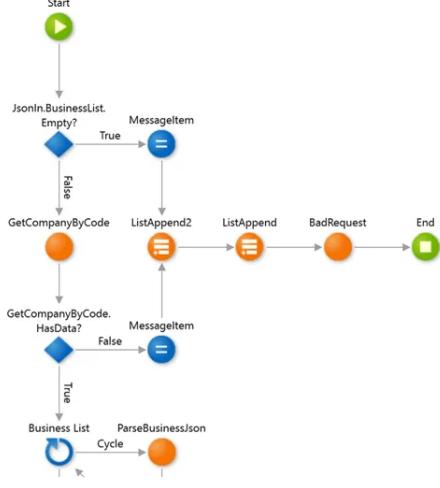


Figura 13: Diretriz 1 — Sentido descendente é progresso

Fonte: [40]

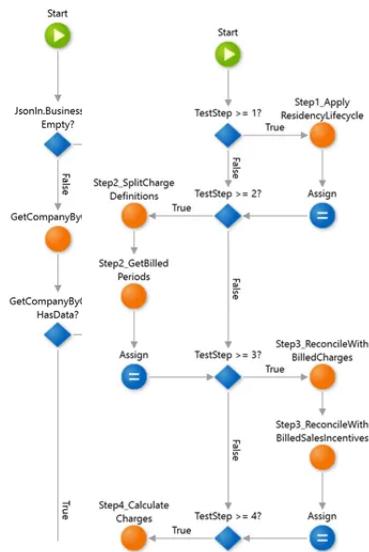
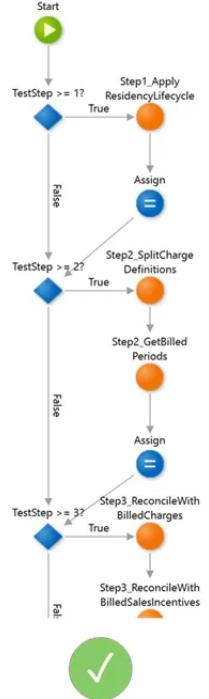


Figura 14: Diretriz 2 — Ramificações para a direita

Fonte: [40]



**Diretriz 3 — Exceções para a direita:** Figura 15, exceções para a direita, mesmo que não seja o ramo “mais importante”. Se se aplicasse a diretriz 2 às exceções, acabava-se com um código maioritariamente horizontal, desta forma impede-se que a exceção se destaque, e mantém-se o foco no código principal;

**Diretriz 4 — Ciclos:** Figura 16, os ciclos devem iniciar-se numa diagonal para cima e para a direita;

**Diretriz 5 — Evitar setas sobrepostas:** Figura 17, as setas sobrepostas devem ser evitadas mesmo que implique ignorar outra diretriz, se for necessário deve-se criar nodos “dummy” para direcionar as setas de forma a que não se sobreponham;

**Diretriz 6 — Alinhar com intenção:** Figura 18, alinhar intencionalmente os nodos relacionados entre si. Por exemplo, se dois ramos têm código similar, os nodos iguais devem estar alinhados horizontalmente para ser ver que será o mesmo código;

Com estas 6 diretrizes é mais fácil organizar e ler o código da plataforma, e permitirá ter um projeto mais homogéneo e previsível[40].

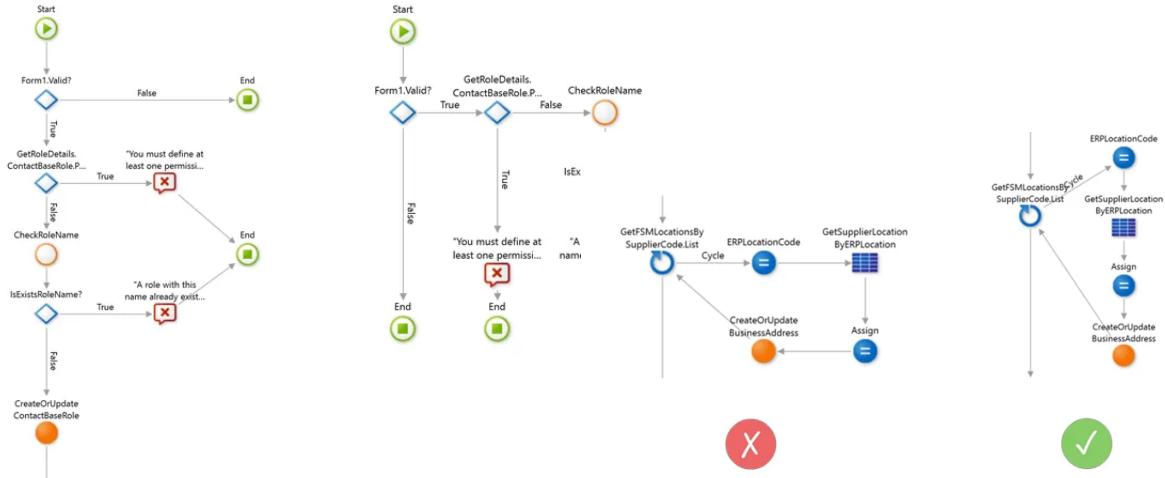


Figura 16: Diretriz 4 — Ciclos

Fonte: [40]

Figura 15: Diretriz 3 — Exceções para a direita

Fonte: [40]

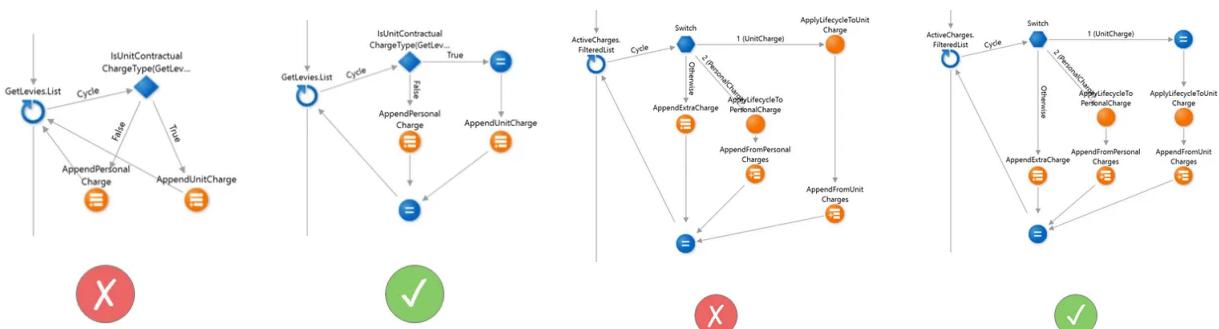


Figura 17: Diretriz 5 — Evitar sobreposições

Fonte: [40]

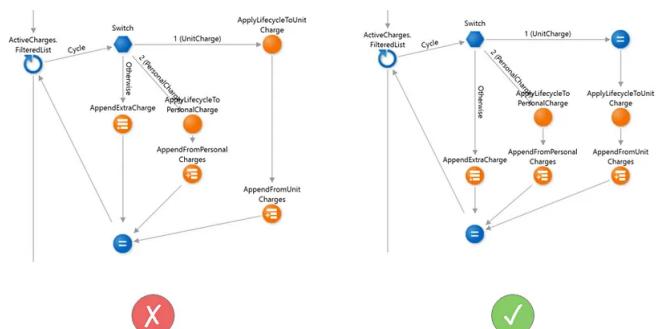


Figura 18: Diretriz 6 — Alinhar com intenção

Fonte: [40]

### 6.1.5 Competidores

No ecossistema de desenvolvimento *low-code*, existem várias plataformas que competem com OutSystems. A escolha entre estas dependerá das necessidades específicas de cada projeto, das preferências da equipa e das metas da empresa.

Apesar de OutSystems ser uma das plataformas mais maduras e bem desenvolvidas para este fim no mercado, podemos fazer uma análise dos seus pontos fortes e fracos:

#### **Pontos Fortes:**

- O Controlo de versionamento pode ser integrado com Git e plataformas como GitHub ou GitLab;
- Várias opções para integração com outras plataformas com ferramentas como o [Integration Builder](#), desde servidores SQL ao MS SharePoint Online;
- Vastas livrarias, pré-definições, e templates disponíveis;
- Ênfase também no desenvolvimento de aplicações móveis.

#### **Pontos Fracos:**

- Uma personalização de UI menos flexível em comparação com algumas alternativas;
- Flexibilidade para correr os próprios servidores está apenas disponível em alguns planos;
- Colaboração não é um dos aspetos mais fortes da plataforma;
- Custo comparativamente elevados em relação a outras alternativas[\[41\]](#).

Na Figura 19 pode-se ver algumas das plataformas *low-code* mais usadas nos últimos cinco anos de acordo com o Google Trends e na Figura 3 como se comparam com as demais.

O ambiente de desenvolvimento *low-code* é muito competitivo, pelo que existe uma panóplia de ferramentas com objetivos similares, como por exemplo: Bonitasoft, Appsheet, Bubble, Zoho Create, Budibase, Backendless, Eclipse/SAP, Salesforce Lightning, etc. Pode ser que nenhuma ferramenta ou combinação entre as ferramentas aqui analisadas seja a mais indicada para qualquer projeto. Algumas equipas também adotam abordagens com uma pilha de infraestrutura constituída por vários programas de ecossistemas diferentes, utilizando diferentes ferramentas para aspetos específicos, como Webflow para o front-end e Zapier ou N8n para o lado do servidor. No fundo, esta área é muito flexível e está sempre em constante evolução, e pode ser que a infraestrutura mais indicada para um projeto seja uma que tenha que ser feita à medida.

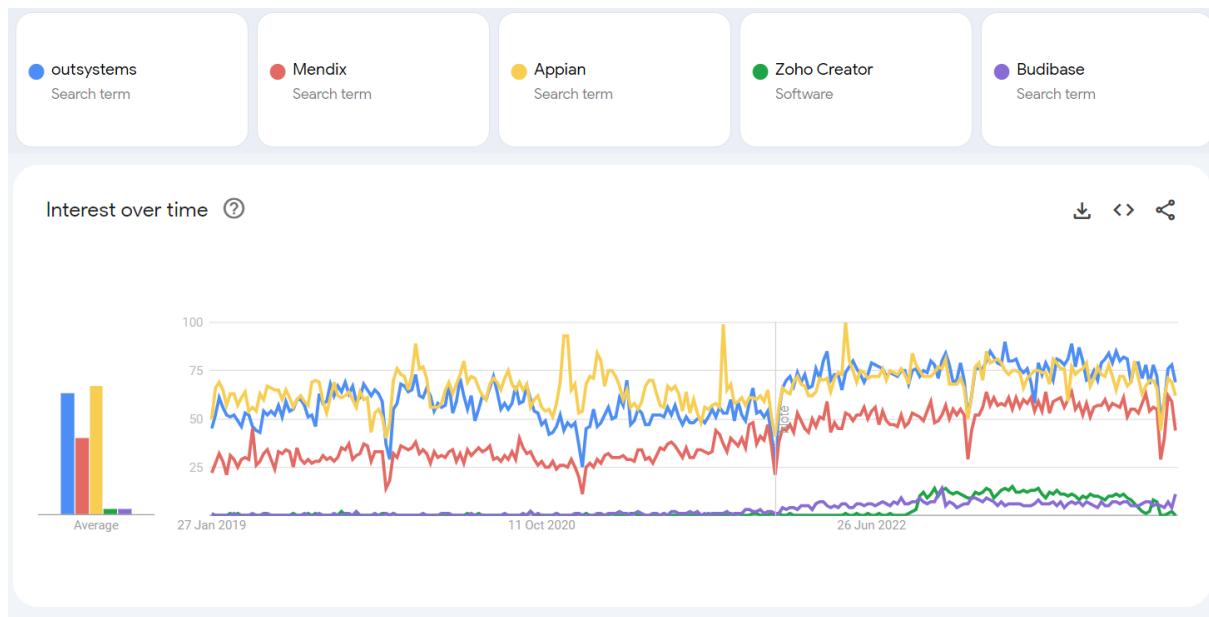


Figura 19: Google Trends for Low Code Platforms Worldwide

Fonte: [42]

Tabela 3: Comparação de Plataformas Low-Code

Fonte: [41], [43]

Campo	Plataformas			
	OutSystems	Mendix	Appian	Pega
Facilidade de Utilização	Médio	Alto	Alto	Baixo
Integração	Muito Alto	Alto	Alto	Alto
Escalabilidade	Muito Alto	Muito Alto	Muito Alto	Muito Alto
Foco em Desenvolvimento Móvel	Alto	Alto	Alto	Alto
Colaboração	Alto	Muito Alto	Alto	Alto
Custo	Alto	Disponível após Pedido	Alto	Disponível após Pedido

### 6.1.5.1 Mendix

Fundada em 2005, a Mendix é uma plataforma de desenvolvimento *low-code* projetada para fornecer às empresas ferramentas para rapidamente construir e distribuir aplicações personalizadas. Os fundadores acreditavam que o desenvolvimento e a distribuição de software iria beneficiar bastante com uma mudança de paradigma, e daí nasceu a empresa. Mendix foca-se na colaboração permitindo às equipas trabalhar efetivamente em conjunto e simplificando o desenvolvimento através de modelagem visual com uma interface intuitiva, como visível na Figura 20, e componentes reutilizáveis[44].

O Mendix destaca-se pela sua flexibilidade enquanto mantém um bom nível de intuição. Oferece também um mercado privado de aplicações que permite a sua partilha mesmo internamente. Proporciona uma variedade de opções de implementação, incluindo diferentes serviços cloud, soluções locais e híbridas, adaptando-se sempre às diversas necessidades da organização[41].

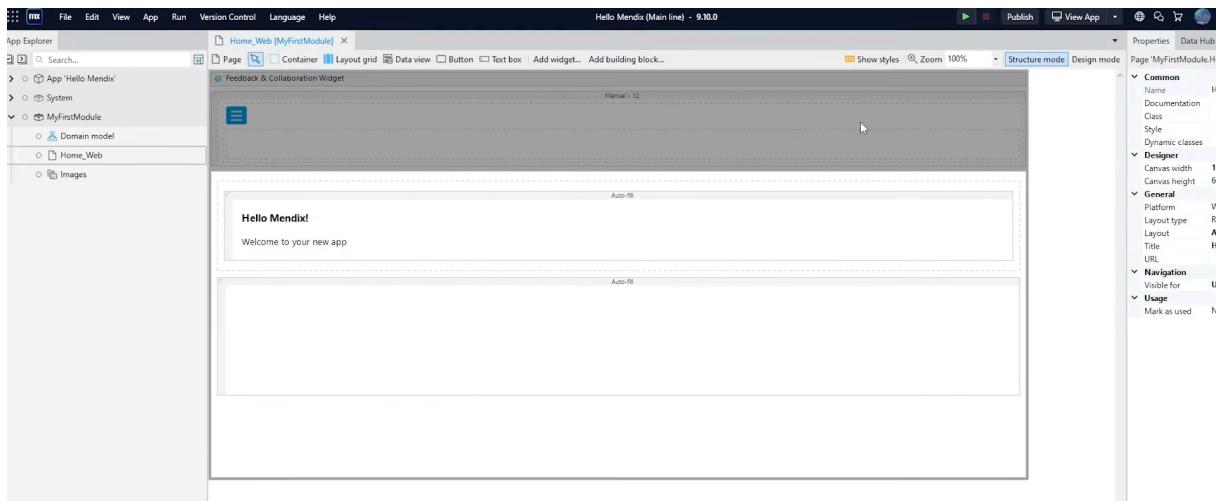


Figura 20: Mendix UI

Fonte: [45]

### 6.1.5.2 Appian

Fundada em 1999 por Michael Beckley, Robert Kramer, Marc Wilson e Matthew Calkins e com uma interface como observável na Figura 21, a Appian é uma plataforma de desenvolvimento *low-code* criada com o objetivo de simplificar e acelerar o processo de desenvolvimento de software e com a visão de que pessoas talentosas e com paixão, quando proporcionadas com as ferramentas certas irão dar resultados incríveis[46], [47].

Appian destaca-se no mercado de desenvolvimento de software devido a várias funcionalidades distintas. Algumas delas incluem:

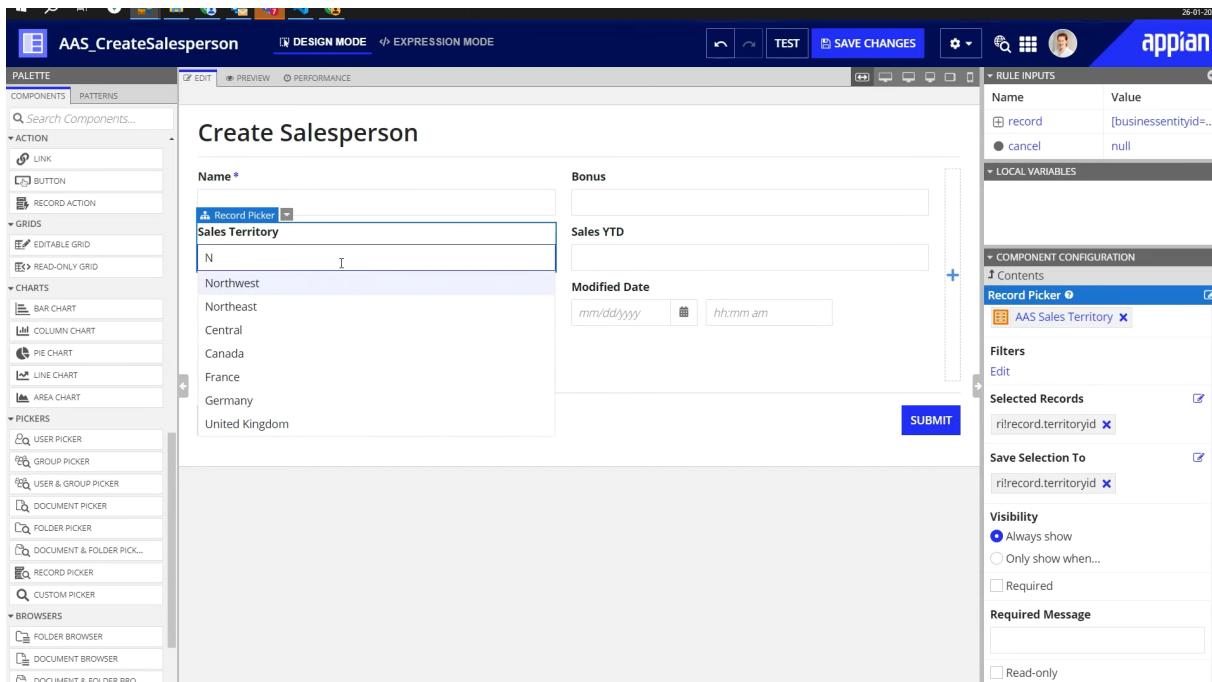


Figura 21: Appian UI

Fonte: [45]

### 1. **Deployment Eficiente:**

- Permite a implantação rápida de aplicações com variadas ferramentas de implantação;
- Oferece a sua própria solução para controlo de versionamento.

### 2. **Vários Formatos para Apresentação de Dados:**

- Possibilita apresentar dados em vários formatos, como tabelas, gráficos em formato grade e PDF.

### 3. **Desenvolvimento Visual:**

- Programadores podem moldar fluxogramas para criar diagramas de processos de forma visual, eliminando a necessidade de codificar manualmente.

### 4. **Pontos a Considerar (Desvantagens):**

- Processo de documentação precisa de ser aprimorado;
- Limitações na integração com outros produtos;
- A ferramenta de gestão de permissões pode ser um pouco lenta;

- Desenvolvimento geral de aplicações pode ser mais lento em comparação com as alternativas analisadas.

Appian destaca-se na eficiência da implantação de aplicações e na criação de processos através de fluxogramas visuais. A sua flexibilidade na apresentação de dados, em junção com a capacidade de desenvolver aplicações para diversos setores, fazem dela uma opção bastante versátil[43].

#### 6.1.5.3 Pega Platform

A Pega Platform, a plataforma mais antiga das analisadas, fundada em 1983, foi criada por Alan Trefler, um programador na área financeira e dos seguros, fundou Pega devido à desconexão que existia entre os métodos tecnológicos usados e os processos e objetivos empresariais. Desde então, a plataforma tem desempenhado um papel crucial no cenário de desenvolvimento de software, fornecendo soluções robustas para diversas necessidades empresariais, a sua interface está representada na Figura 22.

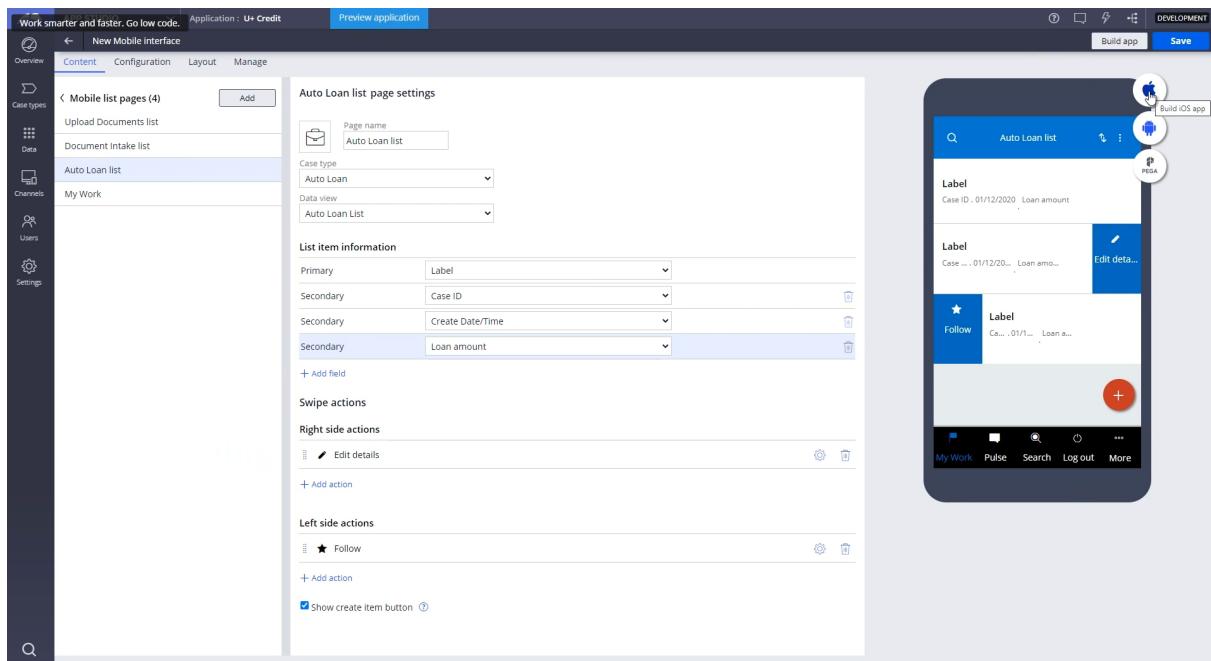


Figura 22: Pega UI

Fonte: [48]

Algumas vantagens e desvantagens da plataforma incluem:

- **Vantagens:**

- Oferece uma abordagem eficiente na gestão de casos, utilizando um modelo cen-

tral que permite controlar tarefas, contribuindo para uma integração forte com os paradigmas das empresas[49];

- Forte capacidade de incorporar e utilizar inteligência artificial nas aplicações.

- **Desvantagens:**

- Tem uma curva de aprendizagem mais acentuada.

## 6.2 ServiceNow

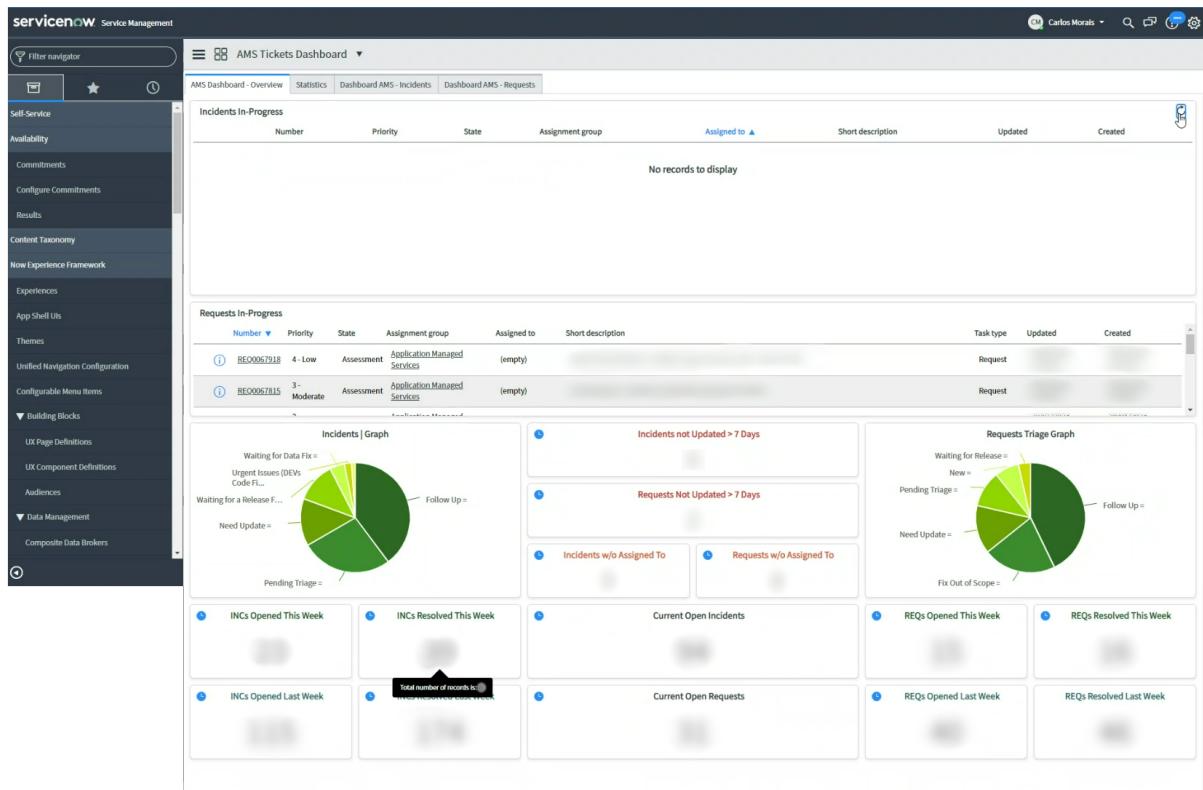


Figura 23: Plataforma ServiceNow - Página Principal<sup>9</sup>

Fonte: Plataforma Interna da ServiceNow

A ServiceNow é uma das plataformas de referência na gestão de serviços empresariais, oferecendo uma ampla gama de funcionalidades para otimizar e automatizar processos internos. Destaca-se pela sua capacidade de integrar e gerir eficientemente serviços de TI, recursos humanos, gestão de projetos e muito mais, a página principal da aplicação está demonstrada na Figura 23. Fundada em 2004 pelo antigo CTO da Peregrine Systems, Fred Luddy, ServiceNow iniciou a sua existência num único computador portátil, numa pequena sala de escritório, apoiada por um único colaborador e um punhado de voluntários dedicados.

<sup>9</sup>Alguma da informação nesta e em próximas imagens será censurada devido a questões de confidencialidade.

No seguimento deste começo, Luddy propôs-se a criar uma plataforma baseada na nuvem para encaminhar o trabalho de forma eficaz, sendo suficientemente poderosa para impulsionar o sucesso empresarial, mas, simultaneamente, simples o suficiente para não afastar potenciais utilizadores. Não é surpreendente que este objetivo tenha chamado a atenção dos clientes. A empresa cresceu rapidamente e, em 2012, tornou-se pública.

Desde então, ServiceNow expandiu significativamente as suas ofertas, e as mesmas prioridades que definiram a sua fundação há vinte anos continuam a defini-lo hoje: uma paixão por fazer as pessoas tirarem mais proveito da tecnologia.

Atualmente, as ofertas da ServiceNow incluem gestão de serviços de IT, gestão de operações de IT, gestão de serviços ao cliente, gestão de recursos humanos, operações de segurança, risco e conformidade, entrega de serviços no local de trabalho e gestão de serviços de campo - tudo numa só plataforma.

É através da ServiceNow que os utilizadores reportam os dois tipos de incidentes, request ou incidente, como explicado na secção 3.3. De seguida estes passam por uma equipa de suporte que tenta filtrar e resolver os incidentes mais simples, como utilizadores a utilizar mal a plataforma, ou por vezes incidentes que a equipa de desenvolvimento os instruiu a resolver, por exemplo, se um certo problema aparecer, instruam os utilizadores a apagar a cache do browser. Após esta fase, se não foi possível resolver, são enviados para a ServiceNow onde são analisados pelo team leader da equipa em triagem, que manda uma mensagem padrão para o utilizador a pedir desculpas pela inconveniência e a informar que se está a investigar o problema, e prioriza os incidentes da seguinte forma:

- **Low** - Um erro de baixa importância que não bloqueia o utilizador, como uma desformatação do site;
- **Medium** - Quando é mais urgente no sentido de poder afetar a funcionalidade da plataforma, mas haver como contornar o problema, como recarregar a página ou refazer o contrato;
- **High** - Quando o utilizador está bloqueado e não consegue aceder à funcionalidade que necessita. Esta prioridade também pode ser dada a utilizadores de certas organizações maiores, no sentido de serem clientes mais importantes que a empresa quer garantir que mantém;
- **Critical** - Prioridade atribuída a erros urgentes, como erros 502, quando o site está em baixo e ninguém conseguir aceder.

Seguidamente, ao longo da vida de um incidente este será movido entre um dos seguintes estados, como se pode ver na Figura 24:

- **New:** Novo — O estado inicial de um incidente que acabou de ser recebido;
- **Pending Triage:** Pendente de Triagem — Aguarda avaliação;
- **Waiting for Data Fix:** Aguardando Correção de Dados — À espera que o *datafix*, pedido à equipa da base de dados, seja efetuado;
- **Urgent Issues (DEVs):** Problemas Urgentes (DEVs) — Incidentes críticos que requerem atenção imediata dos programadores;
- **Need update:** Necessita de Atualização — Requer atualização ou informação adicional como por exemplo o UMR (Unique market reference) para identificar o contrato problemático do utilizador;
- **Follow up:** Acompanhamento — À espera de feedback do utilizador para saber se o problema foi resolvido;
- **Follow up second strike:** Acompanhamento segundo golpe — Segundo a política da empresa, após vinte e quatro horas volta-se a mandar mensagem ao utilizador e se nas vinte e quatro seguintes ainda não houver resposta, fecha-se o incidente;
- **Waiting for release:** Aguardando Lançamento — Foi criado *defect* a partir do incidente e espera-se que seja lançada a versão com a correção para produção.

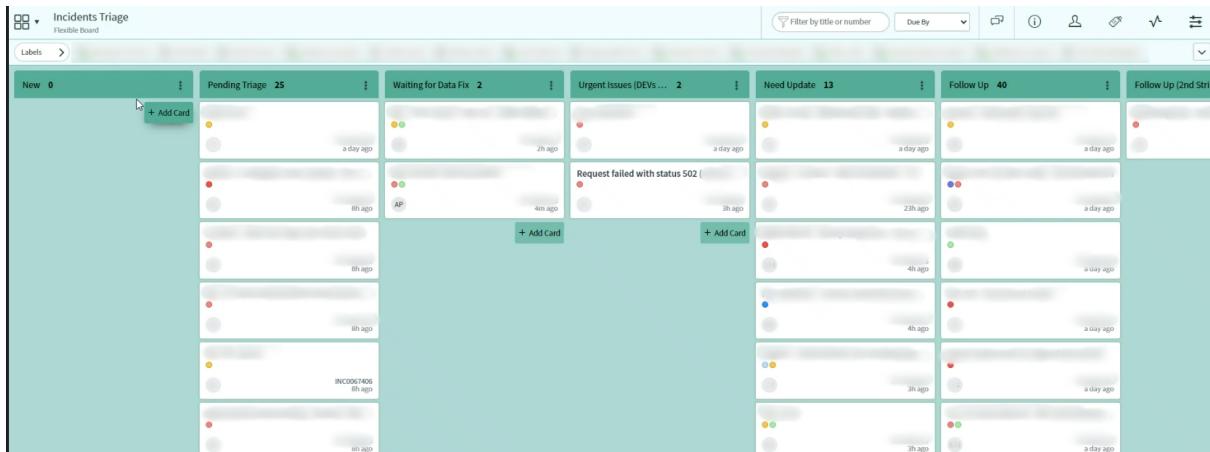


Figura 24: Plataforma ServiceNow - Incidentes

Fonte: Plataforma Interna da ServiceNow

### 6.2.1 Competidores

Neste ambiente altamente volátil e de alta competição no desenvolvimento de software empresarial existirão sempre ferramentas que podem substituir funcionalidades de outras, no entanto, no

caso da ServiceNow é difícil encontrar uma que faça exatamente o que esta plataforma faz.

Na realidade o que se alcança com a plataforma ServiceNow pode ser já alcançado pelas ferramentas usadas como o Jira ou Confluence ou por outras como ClickUp[6.6.3.2], mas a verdadeira vantagem da ServiceNow é a habilidade de ter tudo no mesmo sítio, sendo por isso uma ferramenta única na gestão empresarial.

## 6.3 Plataforma Atlas

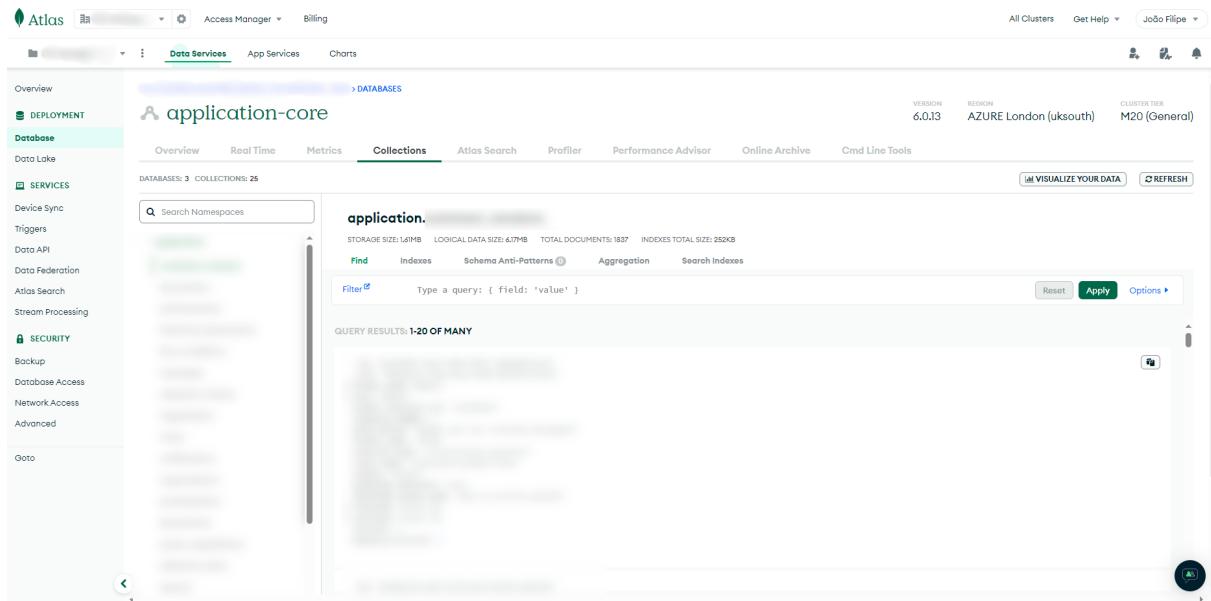


Figura 25: Plataforma Atlas

Fonte: Plataforma Interna do Atlas

Atlas, cuja interface se pode ver na Figura 25, é um componente crucial do processo, e é uma das plataformas mais usadas no que toca à resolução e análise de problemas.

A plataforma está integrada com todos os ambientes do projeto, permitindo escolher entre estes no canto superior esquerdo. Existem, no entanto, algumas restrições, por exemplo, para se poder ter acesso ao ambiente de produção, é necessário aceder primeiro à [Plataforma Azure](#), aceder a “My roles” > “Groups” e Pedir acesso com um campo para mencionar o número do Incidente em produção que vai ser analisado e uma pequena descrição.

Um dos campos mais usados na plataforma é o dos filtros, visível na Figura 25 onde é possível filtrar por campos específicos, organizar por data, colocando no campo “sort” por exemplo {“\_metadata.modified\_date”: -1} ou usar “Aggregators”, na aba relacionada que permite a criação de pesquisas complexas passo a passo, removendo alguma dessa complexidade.

É também aqui onde se gerem algumas das permissões relacionadas com a Base de Dados, ações reservadas apenas a alguns profissionais por questões de segurança.

### 6.3.1 Competidores

Em vez de uma base de dados NoSQL, a plataforma poderia beneficiar de uma base de dados SQL, até devido à melhor integração com outras ferramentas e à maturidade do mercado.

Pelo que uma das alternativas a baixo tratadas é uma base de dados SQL.

#### 6.3.1.1 Cosmos DB

Visto que é parte do ambiente Azure que já é extensivamente utilizado, seria mais fácil integrar com o projeto. Isto não quer dizer que uma transição seria fácil, normalmente transições deste tipo têm que ser muito bem justificadas devido aos preços e mão de obra exigidos.

A escolha entre Cosmos DB e MongoDB depende das necessidades específicas da empresa. Cosmos DB destaca-se quando se necessita de flexibilidade com múltiplos modelos de dados, enquanto que o MongoDB se concentra no modelo de documentos. Cosmos DB oferece escalabilidade global e replicação automática em várias regiões, tornando-se ideal para baixa latência e elevada disponibilidade[50].

#### 6.3.1.2 MariaDB

Esta seria uma solução que manteria um paradigma NoSQL como o MongoDB.

MariaDB, uma base de dados relacional, adota o formato de tabela testado pelo tempo. Cada tabela no MariaDB segue um esquema fixo, definindometiculosamente as colunas e os respetivos dados. Esta forma metódica de armazenar dados não apenas garante precisão, mas também estabelece estruturas predefinidas, fomentando clareza e organização o que contrasta com a utilização mais relaxada de BSON para armazenar dados em MongoDB.

## 6.4 Plataforma Azure

A plataforma Azure é a que engloba mais operações e áreas distintas do projeto, desde segurança até à gestão das máquinas virtuais usadas para os ambientes.

Existe uma panóplia enorme de serviços, como é visível na Figura 26, estes incluem “logic apps”, “azure storage”, “storage accounts”, “Azure Virtual Network” e “Azure Health”, etc. Alguns dos que são mais usados e geridos na plataforma são os seguintes:

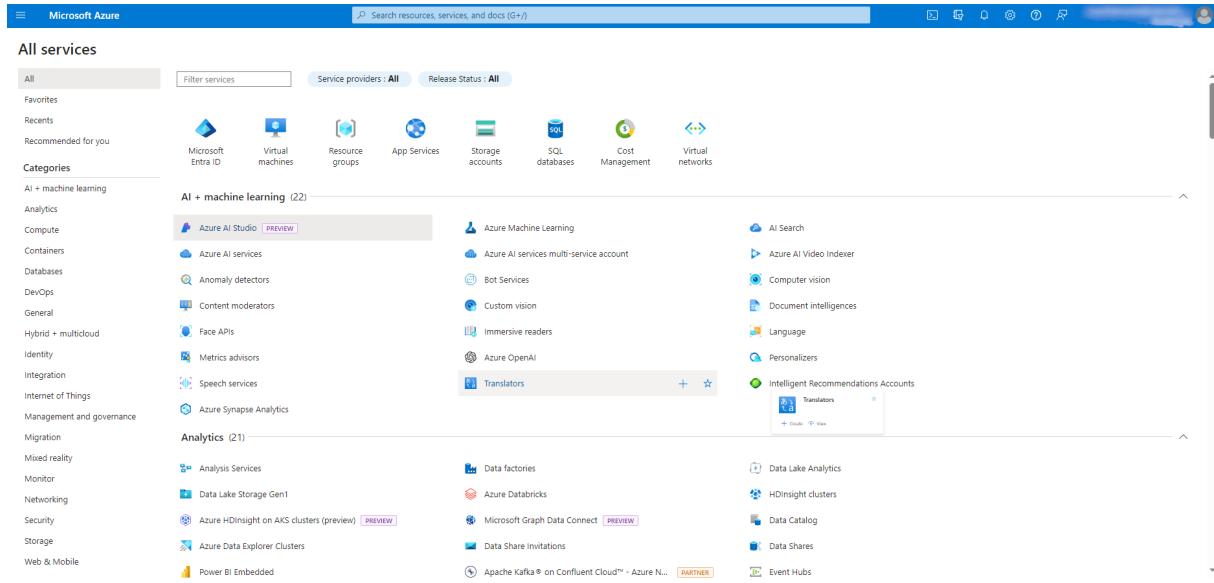


Figura 26: Plataforma Azure

Fonte: Plataforma Interna Azure

#### 6.4.1 Resource Groups

Os *Resource Groups* desempenham um papel crucial na organização e gestão eficiente dos recursos. Permitem agrupar recursos relacionados, como recursos de certos ambientes. Ao utilizar estes grupos, a administração torna-se também mais fácil e permite políticas de segurança e controlo de acessos a grupos específicos.

#### 6.4.2 Virtual Machines

É na gestão de Máquinas Virtuais do Azure que é possível aplicar os conceitos de *scale out* e *scale up* discutidos na Secção 5.1.1, muitas vezes necessário para fazer um *deployment* específico ou devido a fluxos invulgares de utilizadores para a plataforma. É nesta secção que se monitoriza o seu desempenho, se aplicam atualizações e se configuram as definições necessárias.

#### 6.4.3 SQL Database

A secção de SQL Database no Azure é essencial para a gestão e análise, manual ou automática, de bases de dados e outros problemas gerados pelas diferentes plataformas. É aqui que se tem acesso a todas as ferramentas avançadas do Azure para analisar os dados que podem ter vindo de qualquer ponto da plataforma.

#### 6.4.4 Competidores

Devido à magnitude de serviços e funcionalidades que o Azure providencia, é difícil encontrar uma alternativa que disponha de todas as mesmas ferramentas, ou não tenha as suas próprias funcionalidades diferentes. Mas algumas ferramentas que podem ser consideradas são as seguintes:

##### 6.4.4.1 Amazon Web Services(AWS)

O Amazon Web Services (AWS) é uma plataforma de nuvem também bastante usada no mercado, visto que a sua empresa mãe é a Amazon. Oferece uma ampla gama de serviços e soluções para atender às necessidades de organizações de todas as dimensões. Lançada em 2006, a AWS tornou-se rapidamente numa referência para o setor.

A escolha entre Azure e AWS é depende muito das necessidades da empresa, por exemplo, uma empresa que precise de melhor integração com o Windows ou com PaaS, poderá estar melhor com o Azure, enquanto uma que necessite mais de IaaS, poderá estar melhor com a AWS.

Ambas as ferramentas oferecem basicamente as mesmas funcionalidades, mas a AWS mantém-se a líder a termos de adoção global a 33% enquanto o Azure se mantém apenas com 13%[51].

##### 6.4.4.2 Oracle Cloud

A Oracle Cloud é também uma plataforma de nuvem proeminente no mercado, pertencente à Oracle Corporation e lançada em 2016[52], incluindo serviços de computação, armazenamento, bases de dados (podendo servir também como alternativa ao Atlas), inteligência artificial, blockchain, etc. Também com um foco especial em aplicações empresariais.

Tem um plano grátis que permite até acesso ilimitado a uma pequena máquina virtual, atraindo de início muitos pequenos negócios e empreendedores individuais.

É também uma plataforma com serviços IaaS mais em foco, e mantém uma adoção de cerca de 8.43%[53].

## 6.5 Plataforma MS Teams

O Microsoft Teams é uma ferramenta essencial para a colaboração e comunicação eficaz na equipa. É onde acontece a maior parte da comunicação em tempo real por mensagens instantâneas, chamadas de áudio e vídeo, e colaboração em documentos em tempo real, tornando a componente geográfica transparente. É também onde se partilham muitos documentos e vídeos importantes para a organização do projeto, desde documentos Excel, como as férias de cada

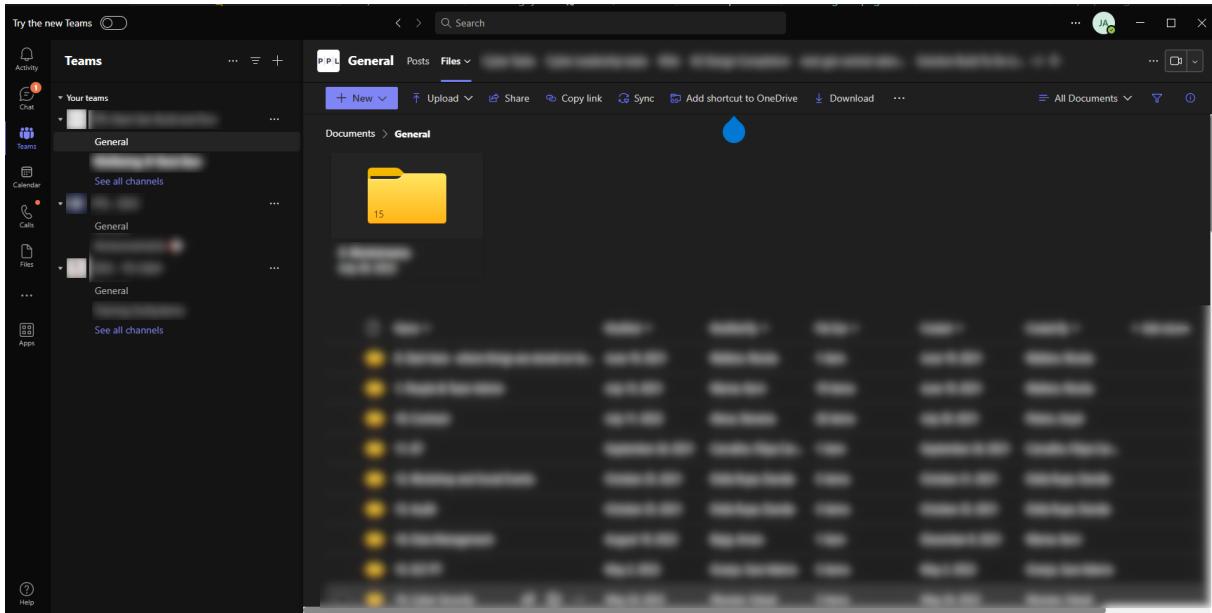


Figura 27: Plataforma do Microsoft Teams

Fonte: Plataforma Interna do Microsoft Teams

contratado, como vídeos gravados de formação dos KTs. Na Figura 27 pode-se ver a página principal com as várias equipas de que um trabalhador faz parte à esquerda.

### 6.5.1 Competidores

Visto que o MS Teams está intimamente integrado com muitas das ferramentas usadas diariamente, como o cliente de e-mail, Outlook, ou o calendário, ou SharePoint para partilha de ficheiros, é preciso considerar alternativas que ofereçam também estas ferramentas, ou permitam uma integração fácil com as mesmas.

#### 6.5.1.1 Slack

O [Slack](#) é uma plataforma de comunicação empresarial, permite a criação de canais para discussões específicas, integração com diversas ferramentas como calendários, permitindo que eventos sejam visualizados diretamente na plataforma. É possível partilhar documentos e ficheiros diretamente, mas não existe uma aba ou sítio onde se possam colocar ficheiros para todos os membros verem, para isto seria necessária uma solução de armazenamento externo como Google Drive ou SharePoint.

Quanto aos preços, o Slack oferece uma versão gratuita com funcionalidades básicas e um histórico de mensagens limitado. As opções pagas, como o Slack Plus e o Slack Enterprise Grid, proporcionam funcionalidades avançadas, armazenamento adicional e suporte.

### 6.5.1.2 NextCloud Stack

O [NextCloud](#) é a melhor alternativa no mercado de código aberto, oferece serviços como calendário, vídeo-chamadas, mensagens, listas *to-do*, partilha de ficheiros, colaboração em ficheiros em tempo real, tudo isto em código aberto e livre de ser vetado, sendo uma boa alternativa para quem queira correr os próprios servidores, havendo também, no entanto, a possibilidade de pagar à empresa para usar os seus serviços na nuvem para correr a plataforma.

Devido à natureza aberta da plataforma, existe uma comunidade vibrante que a rodeia, pelo que existem muitas extensões abertas feitas pela comunidade, havendo uma solução para quase qualquer funcionalidade que possa requerer, mas sendo estas funcionalidades desenvolvidas por indivíduos e não necessariamente equipas com um processo bem definido de desenvolvimento, podem muitas vezes não ser tão robustas como as funcionalidades base da plataforma.

## 6.6 Jira & Confluence

Jira e Confluence são as ferramentas padrão da indústria para gestão de projetos ágeis e documentação colaborativa, respetivamente. A sua popularidade advém não apenas da sua robustez e fiabilidade, mas também da forma como se complementam, em parte por serem ambas desenvolvidas pela mesma companhia, Atlassian, oferecendo uma abordagem holística para o desenvolvimento de software e colaboração em equipa.

Atlassian foi inicialmente fundada em 2002 por Mike Cannon-Brookes e Scott Farquhar como uma empresa que oferecia suporte a outras empresas, mas o software para gestão dos incidentes e defeitos usado na altura não era muito promissor, pelo que desenvolveram a sua própria ferramenta: JIRA. Criaram também um modelo de negócio para o desenvolvimento e venda desta ferramenta e, mais tarde em 2004, desenvolveram também o Confluence<sup>[54]</sup>.

A integração entre Jira e Confluence forma uma poderosa sinergia, esta integração é notável em algumas funcionalidades como: ser possível adicionar incidentes da Jira diretamente na documentação do Confluence que são atualizados automaticamente, criar uma página do Confluence diretamente a partir das páginas da Jira, adicionar o estado de projetos da Jira no Confluence através de dashboards ou templates, ou adicionar novos incidentes diretamente através do Confluence ao sublinhar texto.

### 6.6.1 Plataforma Jira

A Jira facilita o desenvolvimento em metodologias ágeis, e é muito usada no projeto para manter um registo dos *defects* existentes e em que estado se encontram, cada um identificado por um ID específico *PNG-<XXXX>*, juntamente com o ID do incidente de que provieram se tiver sido esse o caso: *INC-XXXXXX*. Existe um fluxo de estados, mostrado na Figura 29, pelo qual cada *defect*

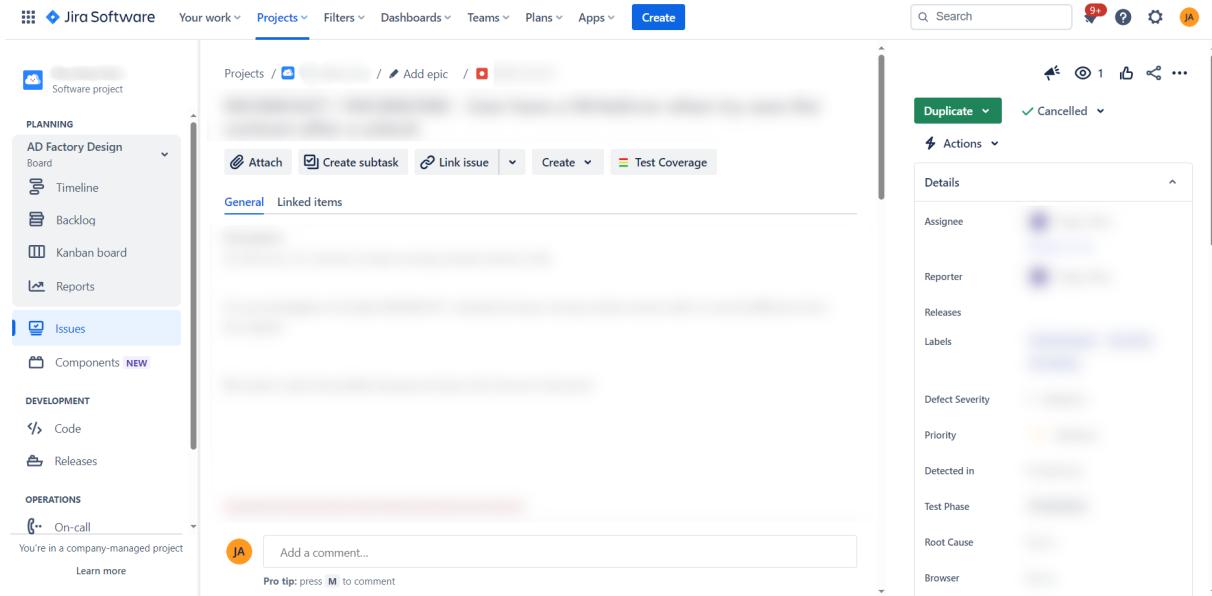


Figura 28: Plataforma Jira Software

Fonte: Plataforma Interna da Jira

tem que passar, começando no estado *New* e acabando em *Closed*, *Duplicate* ou *Rejected*. Na Figura 28 é possível ver interface da plataforma para a visualização de um *defect*.

Em cada *defect* pode-se também adicionar uma lista de *subtasks* a qual necessita de ser completar, muitas vezes designados a diferentes membros da equipa ou até de equipas diferentes, por exemplo: *Merge* do código para um ambiente específico, revisão do código, ou cenários de teste a executar. Cada *subtask* tem também o seu próprio fluxo, como mostrado na Figura 30, começando em *New* e podendo acabar em *Complete* ou *Cancelled*.

É possível também associar outros *defects* na secção *Linked Issues*, e qualquer colaborador pode deixar comentários e mencionar outros colaboradores, que irão subsequentemente receber um e-mail de aviso, permitindo fomentar um ambiente de forte e rápida comunicação.

### 6.6.2 Plataforma Confluence

O Confluence é uma plataforma colaborativa de documentação e é onde a maior parte das descrições dos processos e complexidades está documentada, a sua interface é observável na Figura 31. Gere tudo desde os utilizadores de teste que se podem utilizar nos diferentes ambientes e as suas credenciais, a incidentes ou *defects* que ocorrem frequentemente bem como as suas resoluções, até à documentação de como realizar certas tarefas na aplicação ou numa das plataformas de desenvolvimento.

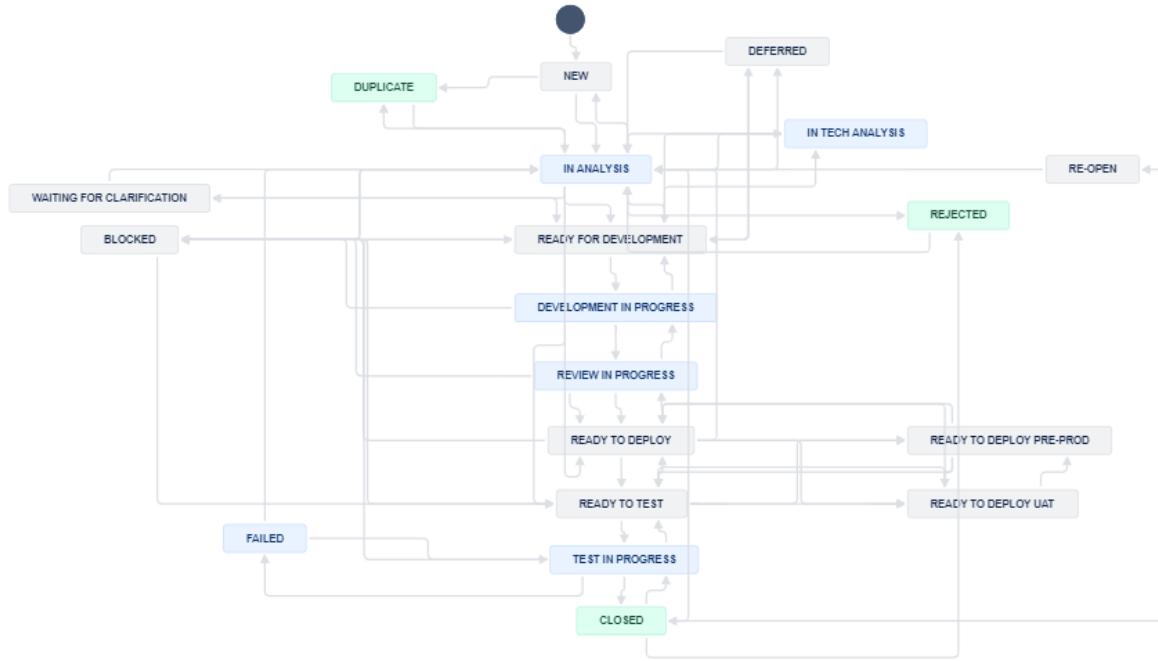


Figura 29: RIL Next Gen *Defect* workflow

Fonte: Plataforma da Jira Interna



Figura 30: RIL Next Gen *SubTasks* workflow

Fonte: Plataforma da Jira Interna

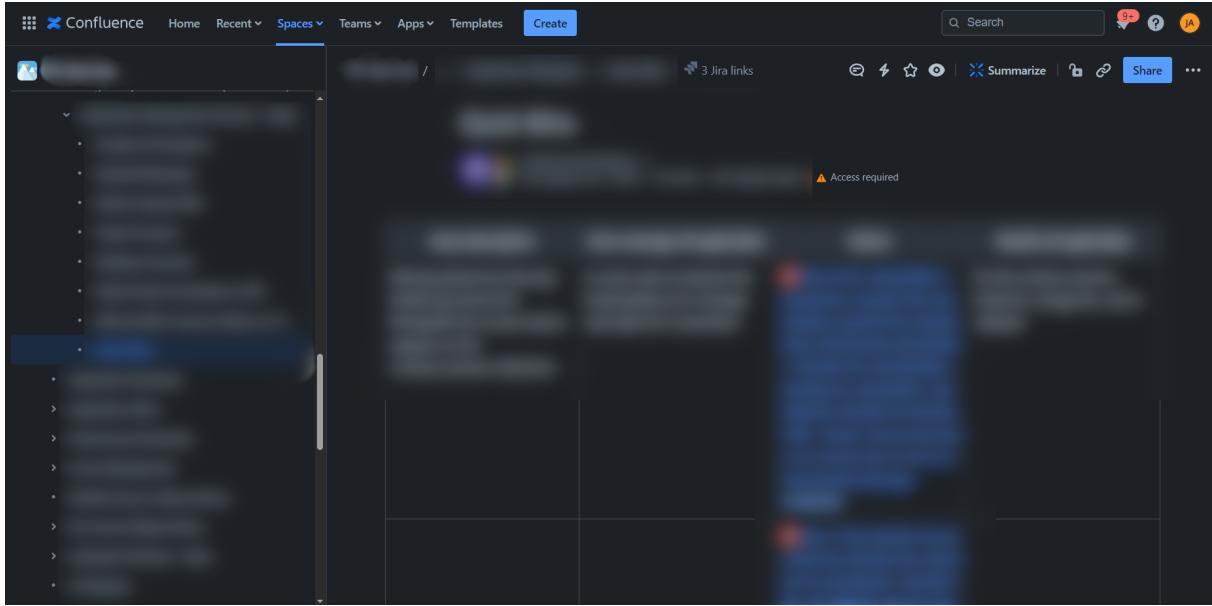


Figura 31: Plataforma do Confluence

Fonte: Plataforma do Confluence Interna

### 6.6.3 Competidores

Apesar de a Jira e do Confluence serem muitas vezes as ferramentas mais indicadas para a gestão de projetos, o preço destas pode ser um fator que leve a considerar outras ferramentas semelhantes, pelo que nesta secção vamos explorar algumas alternativas que possam substituir a junção do Confluence e da Jira.

#### 6.6.3.1 Linear + Notion

[Notion](#) é uma plataforma de colaboração e documentação com uma abordagem flexível e bastante personalizável mantendo uma interface simples e fácil de utilizar, permite a criação de páginas interativas, listas de tarefas, tabelas, integração com outras ferramentas como calendários e neste caso, [Linear](#). Esta última é uma ferramenta para desenvolvimento de software com foco no desempenho e precisão dos processos com funcionalidades como *issue tracking*, listas e *boards* e *cycles*, uma funcionalidade que permite saber sempre qual é a tarefa em que se deve focar de seguida.

#### 6.6.3.2 ClickUp

[Clickup](#) é uma solução que engloba documentação e gestão de incidentes simultaneamente, é uma ferramenta que não teve tanto tempo ainda para se solidificar no mercado, mesmo em termos de integrações com outras ferramentas, mas é uma opção viável para equipas mais pequenas que procurem uma ferramenta mais simples e com um preço de entrada menor[55].

### 6.6.3.3 Bookstack + OpenProject

A alternativa de código aberto para equipas que querem garantir que o código que correm é seguro ou ter a possibilidade de contribuir diretamente para o projeto no caso de faltar alguma funcionalidade importante. Não existe, no entanto, nenhuma integração direta entre ambas as ferramentas.

O [Bookstack](#) é uma ferramenta de gestão de documentação que se destaca pela sua simplicidade e facilidade de uso. Oferece uma abordagem estruturada para a criação e organização de documentos.

Já o [OpenProject](#) é uma plataforma de gestão de projetos de código aberto que oferece uma abordagem abrangente para planear, colaborar e acompanhar o progresso de projetos, permitindo uma gestão eficiente e transparente de tarefas, incidentes e equipas.

## 7 Tarefas no Âmbito do Projeto RIL

Devido à natureza rotativa da equipa, foram efetuadas várias tarefas no âmbito do projeto. Nesta secção vamos analisar alguns dos testes, *defects*, incidentes bem como outras tarefas que tenham sido concretizadas no contexto do estágio académico. Devido ao vasto número de incidentes e defeitos explorados e à natureza efémera da análise destes, visto que frequentemente a análise seria transferida para outra pessoa e ao curto tempo de análise, foi escolhida uma abordagem onde se vai analisar a fundo apenas algumas das tarefas, de forma a se detalhar bastante o processo.

A participação do presente estagiário no projeto inicia-se em simultâneo um com colega, que já colaborava com a Deloitte, mas que era estranho ao projeto. Assim aproveitou-se para ambos aprenderem acerca das ferramentas e da lógica juntos em todas as circunstâncias em que tal fazia sentido.

Como procedimento de acolhimento a recém-chegados, há uma fase inicial, após as formações iniciais mencionadas na Secção 4.2.2, onde novos membros acompanhavam outros membros da equipa nas suas tarefas, como, por exemplo, as tarefas que acompanhavam dependiam das tarefas que estariam a ser realizadas na altura. Com tempo, passariam a fazer estas tarefas individualmente.

### 7.1 Resolver *Defects*

Nesta secção vão-se explorar a fundo alguns dos *defects* resolvidos durante o período do estágio, incluindo ferramentas e métodos de trabalho e análise utilizados. Este método foi escolhido de forma a que o leitor possa ficar com uma ideia clara do que é necessário e importante no processo do trabalho atribuído, desta forma uma análise profunda pode ser executada evitando uma análise mais supreficial de todas. Para uma visão holística do trabalho, a Tabela 4 detalha todos os *defects* trabalhados, respetivas prioridades.

Vamos, primeiramente, detalhar a forma de análise de *defects*:

#### 7.1.1 Análise de *Defects*

De uma forma geral, os passos na análise e resolução de um *defect* devem seguir os seguintes pontos:

1. A análise começa pela reflexão sobre se o *defect* está contemplado, ou não, nas USs, ou seja, se não será um GAP;

Sendo um GAP, é passado à equipa relevante para a US ser alterada, caso contrário avança-se para o próximo passo:

Tabela 4: *Defects* Trabalhados

Fonte: Documentação Interna

Título do <i>defect</i>	Descrição do <i>defect</i>	Prioridade
“A Withdrawn FO showing ‘Requested’ in overview screen & ‘Invalid Data’ error message appearing when clicked on ‘Requested’”	Um <i>broker</i> , ao receber um FO, e esta ser retirada de seguida, o <i>broker</i> consegue visualizar “requested” na tab “overview” para alguns dos UWs. Quando o <i>broker</i> clica no estado “requested”, surge uma mensagem de erro “Invalid Data”.	Baixa
“The broker can’t find the contract using the filter despite the UMR contract is showing in the filter that exists.”	Contratos não conseguiam ser encontrados através dos filtros de procura.	Média
“Regression - Subjectivity email notifications - Wrong emails getting triggered for subjectivity flow”	Quando o UW apaga uma <i>subjectivity</i> com o estado de “satisfaction pending”, a subjetividade passa para o estado “Proposed Deletion”. E quando o Broker aceita a proposta de exclusão da <i>subjectivity</i> , o e-mail/notificação “Firm Order - Update” é acionado em vez do e-mail/notificação “Subjectivity Deleted”.	Média
“Amend Master Facility - UW not visible at Overview tab and Sign and Close page”	O UW não se encontra visível na tab <i>Overview</i> quando se efetua uma alteração à MF.	Média
“PRE - Master Facility - Post Bind Reference or Risk code changes are not showing any stamp details in the transaction logs”	Ao criar um novo contrato, em certas situações, detalhes dos <i>stamps</i> não eram criados. Não foi possível reproduzir.	Média
“Stamp Visibility   Broker - The UW stamps should be filtered by the company.”	Refira à Secção 7.1.3	Média
“Index out of bounds when submiting cancel and replace”	Refira à Secção 7.1.2	Urgente

2. De seguida deve-se identificar o ambiente em que se deve testar e resolver o *defect*;

Existe um campo “Target Fix Release” que indica a release em que o *defect* deverá ser resolvido, esta informação deve ser ligada ao *deployment aggregator* para identificar em qual dos ambientes de desenvolvimento se deve prosseguir, no caso dos ambientes do protótipo da Figura 12, os ambientes DED ou DES.

3. Deve-se então atualizar o “Assignee” para o nosso utilizador na plataforma;
4. Analisar, preparar e corrigir o *defect* na plataforma Service Studio atualizando sempre o estado no *defect* apropriadamente, neste caso de *New* para *In Analysis* para *Ready For Development* conforme o flow demonstrado na Figura 29;
5. De seguida o mais apropriado será alinhar com os maestros. Para tal será necessário realizar testes e revisões do código e possivelmente retrofit para um ambiente mais a baixo, são também criadas “subtasks” para estas tarefas que são, no entanto, frequentemente realizadas por outras equipas.

### 7.1.2 Índice fora dos limites ao submeter *Cancel and Replace*

*Defect* original: “*Index out of bounds when submiting cancel and replace*”

Um dos primeiros *defects* analisados, foi analisado em conjunto com um membro da equipa numa dinâmica de *Remote Pair Programming*.

**Pair Programming:** Também conhecido por *peer programming*, é um estilo de programação ágil onde dois programadores trabalham em conjunto no mesmo dispositivo e no mesmo problema[56]. Existe debate àcerca do método e da sua eficácia, mas tem sido observado, sendo benéfico, em casos em que a tarefa é complexa e uma alta precisão no código é essencial, bem como em tarefas simples onde o tempo é limitado. É também uma boa forma de aprendizagem e integração, onde participantes reportam resultados positivos em áreas como a aprendizagem, confiança no código, tempo necessário e gosto geral no processo[57], [58].

#### Análise e Resolução:

A descrição do *defect* em causa encontra-se na Tabela 5. Fazendo a análise segundo a Secção 7.1.1:

1. Neste caso, visto que o *defect* se referia a um erro de limites que aparecia na própria plataforma, não foi necessário procurar a US referente à funcionalidade, isto porque facilmente se deduz ser um comportamento causado por um erro no código que não é desejado na lógica de negócio;

Tabela 5: Detalhes do defeito '*Index out of bounds when submitting cancel and replace*'

Fonte: Documentação Interna

Defect
<p><b>Descrição do Defeito:</b></p> <p>Ao fazer uma correção do tipo <i>Cancel and Replace</i>, quando o <i>broker</i> submetia a correção ao <i>underwriter</i>, aparecia o erro: “Index out of bounds. Index -1 for bounds [0,0]”.</p> <ul style="list-style-type: none"><li>• Resultado Atual: Ocorre o erro: “Index out of bounds. Index -1 for bounds [0,0]”;</li><li>• Resultado Esperado: O <i>broker</i> deve ser capaz de enviar a correção ao <i>underwriter</i> sem erros;</li><li>• Possível Continuar com o Fluxo: Não;</li><li>• Solução Alternativa: Não há;</li><li>• Passos para Reproduzir:<ul style="list-style-type: none"><li>• Criar um <i>placement</i> com 1 <i>underwriter</i>;</li><li>• Enviar um pedido de Firm Order;</li><li>• Aceitar o pedido de Firm Order;</li><li>• O <i>broker</i> cria uma correção do tipo <i>Cancel and Replace</i>;</li><li>• Não fazer alterações pelo <i>Cancel and Replace</i>;</li><li>• Submeter o pedido pressionando “Send Correction to Underwriter”;</li><li>• Ao seguir estes passos, o erro “Índice fora dos limites. Índice -1 para limites [0, 0]” deve ser exibido.</li></ul></li></ul>

2. Nesta etapa foram feitos testes que demonstraram que o erro se encontrava apenas num dos ambientes de desenvolvimento, pelo que se procedeu a analisar e resolver o defeito neste ambiente;
3. Já se tinha atualizado o “Assignee”;
4. A fase de análise e correção é sempre a mais demorada e imprevisível, sempre atualizando o estado do *defect* quando apropriado:

O processo de desenvolvimento e resolução está muito dependente da comunicação com os membros mais adequados da equipa para cada etapa, pelo que grande parte da capacidade de resolução de problemas, especialmente numa fase inicial, depende muito de uma capacidade de comunicação com os colegas de trabalho.

A análise começou com uma troca de informação com uma BA familiarizada com o flow de *cancel and replace* que nos direcionou para a ação com a lista onde provavelmente ocorria o problema.

De seguida fomos à ação em causa, ativamos o debugger do Service Studio, fizemos um *cancel and replace* na plataforma e identificamos a causa do problema, como é possível ver na Figura 32, o ID do contrato chegava vazio à lista.

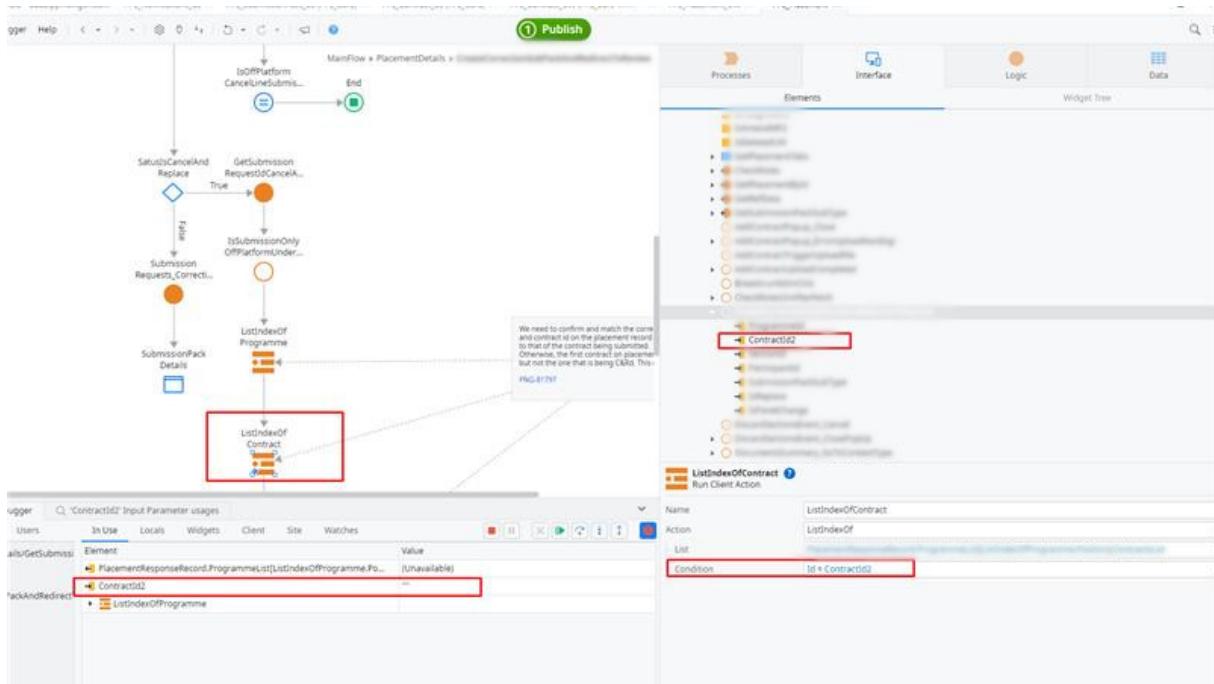


Figura 32: Index Out of Bounds - Root of the Problem

Fonte: Service Studio Interno

Analisou-se então a interface e de onde vinha o ID do contrato inicialmente ao pressionar o botão, e descobriu-se que o ID era passado ao bloco corretamente. Portanto, era no meio da comunicação, desde quando se clicava no botão até à ação em questão, que se perdia o valor.

Os dados eram passados quase exclusivamente de blocos para blocos a partir de *block events*.

**Blocos em OutSystems:** É uma interface reutilizável com código, widgets, ou outros blocos [59].

**Eventos de Blocos:** São eventos que permitem ao bloco interagir com o parente do bloco. Estes eventos podem ter variáveis de input ou de output, visto que o ecrã ou o bloco pai não sabe se algo acontece no bloco, como um botão ser clicado, usam-se eventos para passar esta informação para fora, que podem também ser de caráter obrigatório[60].

Havia uma passagem da informação entre 4 blocos, cujos eventos chamavam diretamente outros eventos, sem haver oportunidade de colocar *breakpoints* no meio do fluxo, depois de uma troca entre um colega acerca do funcionamento de *triggers* em OutSystems, procedeu-se à criação de ações auxiliares que apenas chamavam eventos, e fez-se com que os eventos chamassem estas ações. Desta forma já era possível colocar *breakpoints* e descobrir em que passagem se perdiam os dados.

Veio-se a descobrir que na passagem onde se perdiam os dados, o nome da variável de onde vinham os dados de um bloco era igual ao nome da variável para onde iam os dados noutro bloco, isto normalmente não seria um problema, mas, OS, com o seu código interno em C#, tem dificuldades em fazer um registo e passagem correta de dados em caso de nomes iguais, pelo que se procedeu à mudança do nome e à publicação da aplicação, o que acabou por solucionar o problema.

Retiraram-se então as ações auxiliares e atualizou-se o estado do *defect* adequadamente;

5. Nesta etapa comunicou-se com o nosso líder de equipa, e foi necessário reverter as mudanças. O leitor pode-se ter apercebido que o segundo ponto não seguia o padrão de análise de *defects* estabelecido na Secção 7.1.1. Era necessário analisar o *deployment aggregator* e não apenas resolver o defeito no ambiente de desenvolvimento em que este ocorria.

Veio-se a descobrir que durante aquele período o código de ambas as plataformas de desenvolvimento deveria ser idêntico dado que o código de uma iria substituir a outra brevemente. Pelo que a resolução foi revertida e a análise foi guardada apenas como informação no *defect*, que mais tarde foi usada por um colega que aplicou a resolução na

altura e no ambiente correto.

### 7.1.3 Visibilidade de stamp | Corretora — Os stamps de UW devem ser filtrados pela empresa

*Defect original: “Stamp Visibility | Broker - The UW stamps should be filtered by the company”*

Tabela 6: Detalhes do defeito ‘*Stamp Visibility | Broker - The UW stamps should be filtered by the company*’

Fonte: Documentação Interna

Defect
<p><b>Descrição do Defeito:</b></p> <p>Um <i>broker</i> cria um novo contrato, depois de preencher todas as informações obrigatórias e carregar o documento MRC (documento obrigatório para contratos), seleciona um UW que participará no contrato.</p> <p>Se selecionar um UW A (Empresa X), que pertence a várias equipas dentro da Empresa X e Empresa Y (ambas as empresas estão sob a mesma organização), quando seleciona “Permitted Territory” - “Show All”, só devo ver todos os stamps que o UW A tem atribuídos na Empresa X, mas vê-se os stamps atribuídos na Empresa X e Y.</p> <ul style="list-style-type: none"><li>• Passos para Reproduzir:<ul style="list-style-type: none"><li>• Criar um contrato;</li><li>• Selecionar um UW que pertence a 2 ou mais empresas da mesma organização;</li><li>• Selecionar “Show All” no território permitido;</li><li>• Validar se os stamps disponíveis para o <i>broker</i> são os stamps atribuídos ao UW dentro dessa empresa.</li></ul></li></ul>

#### Análise e Resolução:

A descrição do *defect* encontra-se na Tabela 6. Começou-se a análise pela verificação da US associada à funcionalidade, depois de uma troca com um membro da equipa familiarizado com as USs, identificou-se a US da Tabela 7.

Tabela 7: Detalhes da US “*List of Stamps - Add a Stamp*”

Fonte: Documentação Interna

User Story: Lista de stamps — Adicionar um stamp
<p><b>Visão da User Story:</b> Para a configuração de um Transportador, os stamps estão vinculados à hierarquia (Organização, Empresa e Utilizador). Esta US permitirá a criação de uma lista de stamps. A lista de stamps também incluirá a ocultação de stamps inativos na lista global.</p> <p><b>Como...</b> Administrador RIL</p> <p><b>Eu quero...</b> Adicionar um stamp</p> <p><b>Para quê...</b> Configurar uma lista de stamps</p> <p><b>Critérios de Aceitação de Negócios:</b></p> <p><b>Configuração inicial do ecrã de stamps:</b></p> <ul style="list-style-type: none"><li>• Dado o Administrador RIL estar no ecrã “hierarquia” Quando o Administrador RIL seleciona a aba “stamps” Então, a descrição da aba será “lista de stamps”;</li><li>• Dado o Administrador RIL estar na tela “hierarquia” Quando o Administrador RIL seleciona a aba “stamps” Então, mostra as seguintes colunas de dados (na ordem listada): Nome do stamp, tipo de agência, tipo de stamp, válido a partir de, válido até, estado, classificação e ação;</li><li>• Dado o Administrador RIL estar no ecrã “hierarquia” Quando o Administrador RIL seleciona a aba “stamps” Então, por defeito, mostrar uma lista de carimbos ATIVOS para a organização;</li><li>• <i>etc.</i></li></ul>

Após a análise desta US, deduziu-se que se tratava de um GAP, isto porque na US original dizia “por defeito, mostrar uma lista de carimbos ATIVOS para a organização”, não implicando que devam ser filtrados por empresa. Esta filtragem não se faz na aplicação, e isso torna-se evidente neste caso em que o utilizador pertence a várias equipas de diferentes empresas da mesma organização.

Foi então reportado no próprio *defect* e notificadas as equipas relevantes, que continuaram com o problema a partir daí.

## 7.2 Reportar *Defects*

Pode acontecer que através de testes, ou incidentes, ou mesmo apenas usando a plataforma, se depare com um erro ou comportamento inesperado, o qual pode ter que ser reportado como defeito. Circunstância esta que aconteceu ao presente estagiário quando conseguiu carregar dois documentos MRC para apenas um contrato, algo que não devia ser possível.

Ocorria também, com alguma frequência, membros da equipa descobrirem defeitos durante o apoio a testes. Como parte do seu trabalho cabia-lhes, mediante solicitação, ajudar a documentá-los e a registá-los na plataforma Jira. Na Tabela 8 encontram-se listados todos os defeitos reportados durante percurso do estágio.

O processo de reportar defeitos é bastante intuitivo e envolve a criação de uma entrada no Jira, onde são fornecidos detalhes essenciais para a compreensão e resolução do problema. Este processo implica frequentemente a confirmação com membros da equipa sobre campos cruciais, como a versão principal, tipo de defeito, data de deteção, fase de teste, *sprint*, versão afetada, severidade e prioridade.

Detalhes adicionais, como descrição, passos para reproduzir o defeito, se o utilizador fica ou não bloqueado, qual a solução alternativa, resultado esperado e resultado atual, são também incluídos no registo do defeito. Este nível de detalhe facilita a triagem, análise e resolução eficaz por parte da equipa de desenvolvimento futuramente.

## 7.3 Análise de Incidentes

Incidentes, ao contrário de *defects*, têm uma certa urgência associada, devido a serem um problema reportado diretamente por um utilizador, havendo, normalmente, alguém à espera que este seja resolvido. Pelo que se recorre muito mais a *datafixes* para desbloquear utilizadores e nem sempre é possível ou viável chegar à causa, ou raiz dos problemas. A Tabela 9 apresenta uma lista com todos os incidentes trabalhados.

### 7.3.1 INC0064225 - Todas as seleções são inválidas

Incidente original: “*All selections invalid*”

Tabela 8: Defeitos Reportados

Fonte: Documentação Interna

Título do <i>defect</i>	<i>Defect</i> em Português	Assigned by the end of the in-ternship	Repórter
PNG-89894 Copy Panel Button Appears for a moment in the wrong context	Botão <i>Copy Panel</i> Aparece por um Momento no Contexto Errado	Assigned	João Filipe Silva de Almeida
PNG-88375 Unable to Replace Contract Endorsement document	Não é Possível Substituir o Documento do <i>Endorsement</i> do Contrato	Unassigned	João Filipe Silva de Almeida
PNG-88371 INC0067421, INC0067542 — Adding multiple contract documents to a contract	Possível Adicionar vários Documentos a um Contrato	Assigned	João Filipe Silva de Almeida
PNG-88069 INC0065686-UW able to accept withdrawn request	UW Capaz de Aceitar Pedidos <i>Withdrawn</i>	Unassigned	João Filipe Silva de Almeida
PNG-87070 PROD — "Mark as backloaded"option in contracts shouldn't appear in Archived Placements.	Opção "Mark as backloaded"em Contratos não Deve Aparecer nas Colocações Arquivadas	Assigned	João Filipe Silva de Almeida
PNG-87058 INC0063053, INC0065981 — Endorsement showing "not Responded"	<i>Endorsement</i> Mostrando "Não Respondido"	Unassigned	João Filipe Silva de Almeida
PNG-83536 INC0062203, INC0067349, INC0068073, INC0069140, INC0071112 — UW Reference changes by UW aren't inherited by contract endorsements	Alterações nas Referência feitas por um UW não são Herdadas por <i>Endorsements</i> do Contratos	Assigned	João Filipe Silva de Almeida
PNG-83098 INC0064504 — Regression(Prod) — status withdraw blinks in overview tab	80 Regressão (Produção) — <i>Withdrawn</i> Status Pisca na aba <i>Overview</i> com o <i>Broker</i>	Assigned	João Filipe Silva de Almeida

Tabela 9: Incidentes Trabalhados

Fonte: Documentação Interna

Título do incidente	Incidente em Português	Descrição	Prioridade
Endorsement showing “not Responded”	<i>Endorsement</i> mostrando “not Responded”	<i>Endorsement</i> enviado mostrava “Not Responded” apesar de ter sido respondido, a solução foi apagar e pedir ao utilizador para o recriar	Média
“Unable to submit cancelation”	Não foi possível enviar o cancelamento	Refira à Secção 7.3.2	Média
“Cannot Assign Roles”	Não é possível atribuir os <i>roles</i>	O botão de “Assign Roles” não se encontrava visível. Através da chamada com uma colega e de uma chamada com o utilizador, percebemos que o problema era o botão “Add subpanels” na aba dos <i>Underwriters</i> que estava ativo, e nesse estado, escondia o botão “Assign Roles”	Média
“error NM0011234 to lineslip FNM0003923 — All selections invalid”	erro NM0011234 para <i>lineslip</i> FNM0003923 — Todas as seleções são inválidas	Refira à Secção 7.3.1	Média

Tabela 10: Incidente INC0064225

Fonte: Resumo da Informação do Incidente no Service Now

Incidente INC0064225
<p><b>Descrição do Incidente:</b> O utilizador ficou bloqueado ao submeter uma nova Firm Order. Depois de fazer o processo todo de submissão como esperado, aparecia o erro “All selections are invalid, Please review the breakdown below and select Cancel to return to the Submission Pack”, apresentando por baixo uma lista com todos os UWs a dizer estarem inativos.</p>

A descrição do incidente encontra-se na Tabela 10.

Durante a investigação, verificou-se um possível equívoco no número fornecido pelo utilizador “NM0011234”, que se presumiu que deveria ser “NM0011224”.

Identificou-se inicialmente a possibilidade de resolver o problema removendo e adicionando novamente a *Master Facility* à declaração. Esta abordagem foi sugerida ao utilizador, como uma solução potencial para o problema.

Ao analisar as participações associadas à *Facility*, notou-se ainda uma discrepância numa delas, relacionada com um utilizador chamado “John”. Embora não pareça ser a causa direta do problema, esta discrepância foi registada para investigação adicional, caso a solução inicial não fosse bem-sucedida.

Foi então mandado o pedido ao utilizador para adicionar a MF de novo. Recebeu-se resposta negativa atempadamente, informando-nos que isto já fora tentado e não funcionara.

No entanto, na base de dados era possível ver que a última mudança ao contrato fora há mais de uma semana atrás, foi decidido, portanto, fazer a *datafix* para corrigir o utilizador “John” e marcar uma reunião com o utilizador. Devido ao tempo que o *datafix* levou a ser aplicado, acabou por não se conseguir falar com o utilizador antes do final da semana. Mas mais tarde recebeu-se a notícia que o *datafix* efetuado fora suficiente para desbloquear o utilizador e o incidente pôde ser fechado.

### 7.3.2 INC0065686 - Não foi possível enviar o cancelamento

Incidente original: “*Unable to submit cancellation*”

Tabela 11: Incidente INC0065686

Fonte: Resumo da Informação do Incidente no Service Now

Incidente INC0065686
<p><b>Descrição do Incidente:</b> O utilizador mandou uma Firm Order para um UW que não devia ter mandado, por isso clicou em “withdraw” do pedido, mas o UW conseguiu aceitar. Ao ver que o UW tinha aceite, foi mandado um pedido de cancelamento, mas o UW não conseguia visualizá-lo, e ao abrir a submissão deste pedido, o programa fica parado a carregar infinitamente.</p>

A descrição do incidente encontra-se na Tabela 11.

No decorrer da análise do incidente INC0065686, foi identificado o problema na base de dados. Existe uma sequência de documentos criados na DB de MongoDB que são criados quando o estado de um UW é mudado, estas são as negotiations. Depois de se comunicar com um membro da equipa familiarizado com o funcionamento desta coleção, apercebeu-se que depois da negotiation com o estado “withdrawn” foi gerada subsequentemente uma negotiation com o estado pending\_unconditional\_line, que significa que o pedido tinha sido enviado para o UW mesmo tendo sido withdrawn; se fosse um novo pedido e não o mesmo, teria que haver uma negotiation request\_for\_line\_or\_binder antes da pending\_unconditional\_line.

Não foi encontrada nenhuma indicação na Base de Dados do pedido de cancelamento descrito pelo utilizador.

Depois de alguma discussão decidiu-se que a solução seria fazer um “soft delete” das negotiations depois do withdraw. Para manter a integridade da base de dados foi necessário também alterar a coleção das “participations”, que continha informação àcerca do UW e da sua participação atual no contrato, tendo sido necessário remover cinco campos que tinham sido inadvertidamente adicionados devido às negotiations erróneas que foram criadas, e alterado o campo status da participation para “new”.

No fim foi criado um *defect* com os passos para reproduzir que estavam por definir, onde se detalhou o problema: “UW capaz de aceitar pedido withdrawn”.

## 7.4 Testes

Durante o período de estágio, as responsabilidades incluíam a realização de testes sempre que solicitado pela equipa. O processo de teste envolvia uma abordagem sistemática para garantir o funcionamento e integridade adequados das funcionalidades desenvolvidas, e pode-se dividir nos seguintes passos:

- **Recebimento das Solicitações de Teste:** Ao ser solicitado, forma-se uma compreensão dos requisitos e casos a testar. Como por exemplo a identificação do ambiente a testar, e recolha de quaisquer documentos que possam ser necessários;
- **Recriação das condições do Teste:** Criação das condições na plataforma para recriar o teste com base nas especificações fornecidas;
- **Execução de Testes:** Interação com a aplicação, seguindo as especificações do teste, recolhendo evidências videográficas se necessário;
- **Registo de Resultados:** Documentação dos resultados, por exemplo, no defeito em causa, incluindo comportamentos inesperados e evidências;
- **Comunicação com a Equipa:** Segue-se o relato conciso dos resultados obtidos com quem os solicitou.

A principal destreza necessária para realizar estes testes é uma boa percepção da lógica de negócio e da plataforma. Os seguintes são alguns exemplos de testes feitos durante a duração do estágio:

- Criação de um contrato com duas secções, ambas com uma mistura de UWs off e on platform e uma mistura de *roles* entre eles, mandar e aceitar o contrato, fazer uma *correction* do tipo “panel change” e alterar a percentagem proposta por cada UW sem *roles* de 60% para 50%. Aceitá-las na *section 1* e rejeitá-las na *section 2*. Na *section* rejeitada, deveria voltar a 50% | 50% proposed, em vez de 50% | 60%. O comportamento esperado foi o observado pelo que não se replicou;
- Carregar *placements* arquivados a partir de uma folha Excel onde o *placing broker* especificado pertencia a uma organização com duas companhias diferentes onde ambas tinham *branches* com nomes idênticos. Isto fazia com que ao ir buscar o utilizador do Excel, este pudesse ser extraído do *branch* errado, resultando no bloqueio do utilizador devido à plataforma considerar estes dados inválidos;
- Criar uma *declaration* com um *leader* e um *follower*, nenhum dos UWs têm *roles*, enviar e aceitar, selecionar “cancel line” na *declaration* e enviar. Notar que o *follower* não recebeu notificação da *cancellation*;

- O preenchimento de uma folha de Excel acerca da informação presente no documento gerado pelos contratos e *endorsements* dos contratos numa variedade de casos diferentes com UWs off e on platform.

## 7.5 Análise de Processos

Uma das tarefas delegadas ao presente estagiário foi também a análise de processos de OS. Dentro do Service center é possível analisar nas abas “Monitoring” e “Processes” os processos da aplicação e como é que estes foram concluídos ou em que estado estão, na Figura 33 pode-se ver a interface da plataforma ao visualizar os detalhes de um processo.

Os processos aqui analisados são processos que acabavam suspensos, muitas vezes bastante antigos. São processos que causariam problemas a utilizadores se estes interagissem com o contrato específico associado a este processo, frequentemente podiam ser recomeçados e continuariam normalmente, mas em muitas das situações isto não acontece, esta situação oferece a possibilidade de identificar problemas com a plataforma e possíveis erros a resolver.

Label	Detail	User	Group	Status	Start Date	Close Date	Duration
Upload Files To Mongo		User	Group	Error	12/01/2024 10:27:11	12/01/2024 10:27:11	12/01/2024 10:27:11
Email							

Figura 33: Interface de análise de um processo - Service Center

Fonte: Service Center Interno

Todos os processos revistos, independentemente do seu estado, tinham que ser anotados num documento de Excel específico para a tarefa com informação dos IDs dos contratos, *placements*, equipas e utilizadores associados ao processo, bem como os erros ocorridos, caso tal seja relevante.

Devido à natureza frequentemente repetitiva da análise destes, foi elaborado um web scraper em python que auxiliou na análise de alguns processos, por exemplo, foi automatizado o caso em que o processo `GenerateMRCEmailProcess` estava associado a um utilizador que estivesse inativo, registando e clicando “skip” na plataforma automaticamente, para mais informações refira à Secção [7.5.4](#).

Existem três tipos de processos que geravam erros e era preciso analisar:

### 7.5.1 `SDC_Generation`

Possivelmente o mais difícil de se analisar dos processos aqui representados.

Foi imposto um limite de dez processos por hora e cinquenta por dia dos recomeçados com sucesso, de forma a não sobrecarregar a equipa dos processos.

Os erros levantados neles tinham origem nos stamps das negociações e das suas complexidades. Devido à sua natureza volátil, e a criarem *roles* diferentes para utilizadores e terem uma lógica de pertença a utilizadores e organizações pouco intuitiva, a sua ocorrência dependia de várias contingências.

Em cada processo analisado era necessário encontrar a negociação e os stamps associados e fazer uma análise do campo `sdc_enable` de cada stamp. Caso houvesse pelo menos um com a valor lógico verdadeiro para este campo, tentava-se correr de novo o processo, caso contrário, registava-se e não se tentava correr.

Os processos podiam acabar numa das seguintes formas:

- **Closed:** Quando o processo correu de novo e concluiu com sucesso;
- **Error:** Quando acontecia um erro na execução, o erro era extraído e registado atempadamente;
- **Active — Loop:** Quando fica num estado cíclico, por vezes durante horas. Na maior parte destes casos os processos acabariam por ficar suspensos de novo.

### 7.5.2 `GenerateMRCEmailProcess`

Os processos `GenerateMRCEmailProcess` envolvidos na geração dos documentos MRC (*Mutual Responsibility Contract*). O erro mais recorrente neste processo era identificado pela mensagem “There was a technical issue sending the notification to the user <ID>”. Após uma análise detalhada do fluxo de ações nos logs do Azure, identificou-se que este problema ocorre quando uma ação tenta aceder a um documento na base de dados que não tem o campo URI preenchido, indicando a falta da referência ao documento na plataforma Nuxeo. A solução

recomendada, caso um utilizador se depare e reporte o problema, é solicitar ao utilizador que execute a ação “cancel and replace”.

Outro erro detetado seria o identificado pela mensagem “Invalid Username and password”. Neste caso, era feita uma validação para verificar se o utilizador estava inativo na base de dados. Se estivesse, era pressionado o botão “skip” no processo, pois não havia ações adicionais a serem tomadas. Caso o utilizador estivesse ativo, a informação era anotada e o estado atual era mantido. Eventualmente, poder-se-ia verificar nos logs do Azure as ações executadas até ocorrer o erro para identificar a ação que o causou.

### **7.5.3 UploadDocuments\_V4**

O processo `UploadDocuments_V4` desempenha um papel na gestão de documentos da plataforma, é na sequência de uma asserção da integridade destes documentos na plataforma, que surgiu a necessidade de fazer a análise destes processos.

O principal objetivo é validar se um documento está presente no Nuxeo e na base de dados do OutSystems em sincronia, utilizando o identificador do documento.

O procedimento envolve a consulta à base de dados MongoDB para verificar se o campo URI do documento está preenchido, indicando a existência do documento no Nuxeo, e confirmando na plataforma. Se o documento estiver presente, prosseguia-se com o botão “skip”. Caso contrário, se não existir informação na base de dados MongoDB, a situação seria registada. Nos casos em que o contrato associado ao documento também não é encontrado, a ação “skip” seria considerada. É relevante notar que durante a duração relevante, não foi registado nenhum cenário onde o contrato existia num *placement* mesmo sem ter um URI.

### **7.5.4 Scripts de Automação em Python**

Com o intuito de otimizar a análise dos processos e mediante as adequadas autorizações concedidas, o estagiário assumiu a responsabilidade de automatizar a análise destes. Foram consideradas diferentes opções de interação com os APIs internos do Azure, OutSystems e MongoDB. Contudo, para obter acesso programático à base de dados de produção, seria necessário solicitar aprovação diretamente ao cliente, à RIL. Além disso, os APIs feitos para o Azure de produção não tinham um histórico favorável de funcionamento em Python, existiam apenas programados em C#.

Diante a situação, optou-se por se automatizar através da implementação de um web scraper e da interação direta com o navegador. Os scripts criados estão disponíveis no Anexo [D](#).

#### 7.5.4.1 User Stories

Realizou-se assim um estudo das User Stories (US) a implementar, com o objetivo de reduzir ao mínimo os requisitos, garantir uma implementação rápida e alcançar um elevado nível de automatização. Este processo conduziu à identificação das US das tabelas [12](#), [13](#), [14](#) e [15](#).

Tabela 12: Detalhes da US *Análise de Processos SDC no Estado “Active - Loop”*

User Story: Análise de Processos SDC no Estado “Active - Loop”
<ul style="list-style-type: none"><li><b>Visão:</b> Efetuar uma análise detalhada dos dados dos processos SDC presentes num ficheiro Excel local, identificando aqueles que se encontram no estado "Active - Loop". Posteriormente, realizar uma avaliação dos dados do processo e dos erros associados. A tarefa inclui a atualização do estado para "Suspended" no Excel, juntamente com o registo do tipo de erro e a URL correspondente.</li><li><b>Como...</b> Utilizador autorizado a executar operações sobre os processos SDC no estado "Active - Loop".</li><li><b>Eu quero...</b> Analisar os dados dos processos SDC no estado "Active - Loop", corrigir os erros associados, atualizar o estado para "Suspended" no Excel e registar informações detalhadas sobre o tipo de erro e a URL no mesmo ficheiro.</li><li><b>Para quê...</b> Garantir uma gestão eficaz dos processos SDC, corrigindo problemas associados e mantendo um registo claro das ações realizadas para futura referência e auditoria.</li></ul>

#### 7.5.4.2 Requisitos Funcionais

A partir das USs, foram derivados os requisitos funcionais, apresentados na tabela [16](#). Estes requisitos representam as descrições e comportamentos externos exigidos pelo software, delineando as funcionalidades que o sistema deve cumprir. Proporcionam uma visão de alto nível do sistema.

Na tabela [17](#) está representado, a título de exemplo, um requisito funcional, está acompanhado de um resumo detalhado e descrição. A lista completa dos requisitos encontra-se disponível no Anexo [C](#).

Devido à simplicidade deste projeto, requisitos não funcionais não foram apurados.

Tabela 13: Detalhes da US *Análise de Processos GenerateMRCEmailProcess com Utilizadores Inativos*

<b>User Story:</b> Análise de Processos GenerateMRCEmailProcess com Utilizadores Inativos
<ul style="list-style-type: none"><li>• <b>Visão:</b> Efetuar uma análise dos dados dos processos GenerateMRCEmailProcess, contidos num ficheiro Excel local, focando-se nos processos que possuem apenas o número na primeira coluna, sem informações adicionais. Utilizando a plataforma Atlas, identificar os processos que apresentam o utilizador inativo e, automaticamente, pressionar o botão “skip” na plataforma Service Center para cada um deles. Preencher sempre com os dados do processo no Excel.</li><li>• <b>Como...</b> Utilizador autorizado a realizar operações relacionadas com os processos GenerateMRCEmailProcess e a aceder à plataforma Atlas.</li><li>• <b>Eu quero...</b> Analisar os dados dos processos GenerateMRCEmailProcess presentes no ficheiro Excel, identificar os processos com utilizadores inativos através do Atlas e automatizar a exclusão destes processos na plataforma Service Center.</li><li>• <b>Para quê...</b> Assegurar uma gestão eficiente dos processos GenerateMRCEmailProcess, evitando a interação com utilizadores inativos e otimizando o fluxo de trabalho no Service Center.</li></ul>

Tabela 14: Detalhes da US *Análise de Processos GenerateMRCEmailProcess com Utilizadores Inativos*

User Story: Análise de Processos UploadDocuments_V4 no Excel
<ul style="list-style-type: none"><li><b>Visão:</b> Efetuar uma análise dos dados dos processos UploadDocuments_V4 presentes num ficheiro Excel local. Verificar se o “ContextI” dos processos não tem nenhum documento associado na base de dados de Produção do MongoDB. Em caso negativo, assinalar no Excel com a mensagem “Contract does not exist in Mongo” e prosseguir. Se existir, verificar se todos os documentos possuem um URI e, em caso afirmativo, assinalar no Excel com o comentário “Document in Mongo”. A informação será sempre registada no Excel.</li><li><b>Como...</b> Utilizador habilitado a executar operações relacionadas com os processos UploadDocuments_V4 e a aceder à base de dados de Produção do MongoDB.</li><li><b>Eu quero...</b> Analisar os dados dos processos UploadDocuments_V4 no ficheiro Excel, verificar a existência de documentos associados ao “ContextId” na BD do MongoDB e assinalar no Excel as mensagens adequadas conforme as condições verificadas.</li><li><b>Para quê...</b> Assegurar uma gestão precisa dos processos UploadDocuments_V4, identificando casos em que o contrato não existe na base de dados do MongoDB ou verificando se todos os documentos possuem um URI, proporcionando um registo claro e informado no Excel.</li></ul>

Tabela 15: Detalhes da US *Análise de Processos GenerateMRCEmailProcess com Utilizadores Inativos*

User Story: Interrupção Controlada da Execução do Script
<ul style="list-style-type: none"><li><b>Visão:</b> Permitir a interrupção controlada da execução do script a qualquer momento através da combinação de teclas CTRL + ALT + S.</li><li><b>Como...</b> Utilizador precisa de interromper a execução do script de forma controlada e imediata.</li><li><b>Eu quero...</b> Ter a capacidade de interromper o script em execução em qualquer ponto utilizando a combinação de teclas CTRL + ALT + S.</li><li><b>Para quê...</b> Proporcionar ao utilizador um meio eficiente de parar a execução do script quando necessário, garantindo uma interrupção rápida e controlada, sem comprometer a integridade dos dados ou funcionalidade do sistema.</li></ul>

Tabela 16: Requisitos funcionais dos scripts para análise de processos

#	Nome
REQ-01	Análise de Processos SDC
REQ-02	Análise de Utilizadores Inativos em GenerateMRCEmailProcess
REQ-03	Análise de Documentos em UploadDocuments_V4
REQ-04	Interrupção Controlada do Script
REQ-05	Pintar Colunas com Erros de Roxo para Análise Manual

Tabela 17: Requisito funcional *Análise de Processos SDC*

Nome	<b>REQ-01 — Análise de Processos SDC</b>
Resumo	Este requisito visa permitir a análise detalhada dos processos SDC num ficheiro Excel local, focando-se na identificação e correção dos processos no estado "Active - Loop".
Descrição	O sistema deve possibilitar a identificação dos processos SDC no estado "Active - Loop" num ficheiro Excel local. Após a análise, o estado destes processos deve ser atualizado para "Suspended", e informações detalhadas sobre erros e URLs associadas devem ser registadas no Excel.
Requisitos Relacionados	—

#### 7.5.4.3 Implementação e Arquitetura

Seguidamente, procedeu-se à implementação de três scripts distintos para os três primeiros casos de uso. Desta forma, foi desenvolvido um script para analisar cada tipo de processo: um para a análise dos processos SDC já concluídos, outro para a análise dos processos MRC cujos utilizadores se encontravam inativos, e um terceiro para a análise global dos processos Documents\_V4. Para isto foram utilizadas diversas bibliotecas em Python, incluindo:

- `json`: Utilizada para lidar com a manipulação de dados em formato JSON;
- `pyautogui`: Esta biblioteca foi empregada para automatizar a interação com a interface gráfica do utilizador (GUI) quando tal não era facilmente fazível através do driver selenium;
- `pyperclip`: Desempenha um papel crucial na transferência de dados entre a interface do navegador, mais especificamente, a plataforma Atlas e o código, facilitando a cópia e colagem de dados;
- `selenium`: Utilizado para a automação da interação com navegadores web, o `selenium` permite o controlo programático de um navegador, possibilitando a navegação através de páginas web, preenchimento de formulários, extração de dados, etc;
- `BeautifulSoup`: Essencial para a análise e extração e interpretação de dados a partir de documentos HTML;
- `time`: Importante para gerir pausas e esperas durante a execução do script, embora sempre indesejável, sendo um método que deteta a prontidão dos elementos da página melhor, foi usado como uma alternativa nas situações em que se considerou adequado ou em situações de erro;
- `openpyxl`: Desempenhando um papel crucial na manipulação de folhas de cálculo Excel, o `openpyxl` permitiu a escrita e leitura das informações nos ficheiros Excel utilizados pelos scripts.

#### 7.5.4.4 Testes e Validações

Realizaram-se testes unitários em cada função para assegurar a sua integridade durante o desenvolvimento, garantindo que cada componente do sistema funciona corretamente e em conformidade com os requisitos estabelecidos, proporcionando uma base sólida para a fiabilidade do código, isto é um método natural e uma peça fundamental do desenvolvimento. Através dos testes unitários executados, eram frequentemente encontrados erros nas funções e casos que precisavam de ser prevenidos.

### **7.5.5 Documentação Criada**

Ao término do estágio, surgiu a necessidade de desenvolver uma página no Confluence englobando toda a informação acumulada ao longo do processo de reprocessamento dos procedimentos de produção. Com o intuito de garantir acessibilidade universal na RIL, esta página foi redigida em inglês, assegurando compreensão por qualquer membro da equipa. Detalhando minuciosamente o processo de análise de cada procedimento, desde a ativação das permissões de acesso até à configuração do VPN de Azure, a documentação oferece uma visão abrangente e detalhada do percurso seguido.

Foi também realizada uma reunião no final do estágio entre o estagiário e vários membros das equipas que ficariam encarregues da análise dos processos, isto para transferir a informação de forma direta e célere. Esta iniciativa, juntamente com a documentação adicionada, visou consolidar o conhecimento adquirido, facilitando o trabalho de futuros analistas e contribuir para a partilha eficiente de informações na organização. Foi criada também uma secção com uma explicação do uso dos scripts criados, estes foram adicionados aos ficheiros relacionados com processos de produção.



## 8 Conclusão

Neste capítulo, vai ser realizada uma reflexão sobre o trabalho desenvolvido durante o estágio. Vão ser destacados desafios encontrados e avaliados os ganhos do estágio em relação ao que fora planeado inicialmente. Será também explorado o trabalho futuro do projeto e as direções este pode tomar.

O projeto permitiu contribuir para a resolução de mais de sete *defects*, realizar a análise e reportagem de oito novos *defects* da plataforma, colaborar na resolução de pelo menos quatro incidentes, conduzir uma variedade de testes e procedimentos para reprodução de *defects*, o que auxiliou na solução de outros problemas, podendo estes envolver defeitos ou incidentes. Isto envolveu o uso de tecnologias como OutSystems e as suas diversas plataformas, como o Service Center e o Service Studio, bem como linguagens como o SQL e o MongoDB Query Language (MQL) e a plataforma Azure. Além disso, permitiu o reprocessamento e a análise de processos de produção, bem como a sua automação, possibilitando um estudo mais aprofundado dos APIs da plataforma e uma experiência prática com tecnologias como Excel e Python.

### 8.1 Desafios encontrados e melhorias

Ao longo do estágio, foram surgindo diversos desafios que exigiram uma abordagem proativa para serem superados de forma eficiente e atempada. Estas dificuldades, apesar de representarem obstáculos, proporcionaram valiosas oportunidades de aprendizagem e crescimento profissional.

Entre os desafios enfrentados, podem destacar-se os seguintes:

#### **Desafios de Adaptação:**

A adaptação ao projeto é uma constante, devido à equipa de grande escala e ao constante e célere desenvolvimento da plataforma, é impossível estar confortável com todas as roldanas em funcionamento no projeto. Realçam-se, no entanto, os seguintes como sendo os obstáculos mais notórios, especialmente no início do estágio:

1. Adaptação a novas tecnologias;
2. Adaptação aos processos de um grande projeto;
3. Adaptação ao ambiente de uma grande empresa;

#### **Desafios de Comunicação:**

A comunicação eficaz e partilha de ideias com uma, e mesmo, várias equipas que trabalham em conjunto é uma destreza difícil de se ensinar, pelo que o estágio oferece uma grande vantagem

na possibilidade de desenvolver estas capacidades. Estas foram, e permanecerão sempre, um ponto de melhoria, que fora também mencionado nos check-ins mensais:

1. Comunicação eficaz com a equipa;
2. Confiança na partilha de ideias;
3. Iniciativa em intervenções adequadas e partilha de ideias.

## **Outros Desafios**

Para além dos já delineados, houve também outros obstáculos, nomeadamente:

1. Resolução de problemas técnicos inesperados;
2. Gestão eficiente do tempo em tarefas multifacetadas;
3. Distância entre o lugar de trabalho e instituição de ensino.

A colaboração com membros da equipa, a comunicação eficiente, a orientação e o apoio recebidos tiveram um papel crucial na superação destes desafios. Neste contexto, a experiência proporcionada pelos desafios encontrados durante o estágio contribuiu significativamente para o desenvolvimento profissional, sendo estas competências de comunicação os autores principais na peça de preparação para enfrentar futuros desafios com confiança e determinação.

## **8.2 Reflexões Finais**

Durante o período de estágio, a experiência revelou-se mais educativa e árdua do que inicialmente prevista. A dinâmica entre a plataforma OutSystems e as exigências específicas da empresa de seguros proporcionou uma experiência enriquecedora, repleta de desafios e oportunidades de aprendizagem significativas.

Entre as inúmeras oportunidades que surgiram ao longo do estágio, destacam-se:

1. Aprofundamento do conhecimento da plataforma OutSystems e outras tecnologias. A interação com as diversas tecnologias permitiram uma imersão profunda no desenvolvimento de aplicações ágeis, alinhando-se às necessidades específicas da empresa de seguros e proporcionando uma base sólida de conhecimento nas diversas ferramentas;
2. Colaboração com uma equipa especializada. A oportunidade de trabalhar com uma equipa dedicada e conhecedora do setor de seguros e do setor de *low-code*, possibilitou uma compreensão mais abrangente das complexidades e nuances destas áreas e dos processos de um trabalho empresarial;

3. Resolução de desafios específicos da área. Os desafios enfrentados estiveram diretamente relacionados com as necessidades do setor de seguros, desde a gestão eficiente de dados sensíveis até à implementação e manutenção de processos mais ágeis e eficazes.

### **8.3 Trabalho Futuro**

O projeto atual tem já mais de dois anos com a Deloitte e é a segunda iteração da plataforma, vai sempre, no entanto, necessitar um esforço de manutenção, quer seja para atender a questões e pedidos dos utilizadores, quer para manter uma norma de segurança alta.

O trabalho futuro vai basear-se essencialmente nas *user stories* pedidas pelo cliente e problemas encontrados na plataforma, no entanto, em termos de arquitetura e processos, existem também vários pontos que podem ser melhorados.

**Tarefas planeadas a ser executadas:**

- Criação de um API geral da plataforma de seguros para se poder interagir com esta programaticamente;
- Atualização e manutenção constante de dependências e ferramentas;
- Atendimento às necessidades e pedidos dos utilizadores;
- Melhoria e atualização da documentação.

**Tarefas que poderão ser realizadas:**

- Automatização parcial ou total da análise de processos de OutSystems que acabam suspensos ou em erro;
- Integração com o API em desenvolvimento da plataforma para automatização dos testes.



## Referências

- [1] ISCAC. «Síntese Histórica, 100 anos de Ciências Empresariais.» [https://web.archive.org/web/20231024164219/http://www.iscac.pt/index.php?m=47\\_10&lang=PT](https://web.archive.org/web/20231024164219/http://www.iscac.pt/index.php?m=47_10&lang=PT). (2013), URL: [http://www.iscac.pt/index.php?m=47\\_10&lang=PT](http://www.iscac.pt/index.php?m=47_10&lang=PT) (acedido em 23/10/2023).
- [2] W. contributors. «Data de integração no Ensino Superior Politécnico.» [https://web.archive.org/web/20231024173340/https://pt.wikipedia.org/wiki/Instituto\\_Superior\\_de\\_Engenharia\\_de\\_Coimbra](https://web.archive.org/web/20231024173340/https://pt.wikipedia.org/wiki/Instituto_Superior_de_Engenharia_de_Coimbra). (2004), URL: [https://pt.wikipedia.org/wiki/Instituto\\_Superior\\_de\\_Engenharia\\_de\\_Coimbra](https://pt.wikipedia.org/wiki/Instituto_Superior_de_Engenharia_de_Coimbra) (acedido em 23/10/2023).
- [3] DRE. «Decreto Lei º389/88 - Integra dos Institutos Superiores em Ensino Superior Politécnico.» [https://web.archive.org/web/2020111500000\\*/https://dre.tretas.org/dre/2012/decreto-lei-389-88-de-25-de-outubro](https://web.archive.org/web/2020111500000*/https://dre.tretas.org/dre/2012/decreto-lei-389-88-de-25-de-outubro). (2016), URL: <https://dre.tretas.org/dre/2012/decreto-lei-389-88-de-25-de-outubro> (acedido em 23/10/2023).
- [4] U. de Coimbra. «Universidade mais antiga de Portugal.» <https://web.archive.org/web/20231024175214/https://www.uc.pt/sobrenos/historia/>. (1997), URL: <https://www.uc.pt/sobrenos/historia/> (acedido em 23/10/2023).
- [5] U. de Coimbra. «Associacao de estudantes mais antiga de Portugal.» <https://web.archive.org/web/20231024175846/https://www.uc.pt/driic/rualarga/anteriores/19/19/print>. (2023), URL: <https://www.uc.pt/driic/rualarga/anteriores/19/19/print> (acedido em 24/10/2023).
- [6] Deloitte. «William Welch Deloitte.» <https://web.archive.org/web/20231024180409/https://www.deloitte.com/global/en/about/story/purpose-values/ties-that-bind.html>. (2023), URL: <https://www.deloitte.com/global/en/about/story/purpose-values/ties-that-bind.html> (acedido em 24/10/2023).
- [7] D. UK. «Historia da deloitte em 4 minutos.» <https://web.archive.org/save/https://www.youtube.com/watch?v=SyoLqUP6dX4>. (2013), URL: <https://www.youtube.com/watch?v=SyoLqUP6dX4> (acedido em 24/10/2023).
- [8] E. News. «Big Four.» [https://web.archive.org/web/20231024182741/https://www.euronews.com/2023/09/27/spain-fines-big-four-consulting-firms-for-marathon-working-days#~:text=A%20year%20](https://web.archive.org/web/20231024182741/https://www.euronews.com/2023/09/27/spain-fines-big-four-consulting-firms-for-marathon-working-days)

after%20the%20Spanish, at%20least%20%20%AC1.4%20million. (acedido em 24/10/2023).

- [9] Deloitte. «Deloitte US and our global network». <https://web.archive.org/web/20231024183128/https://www2.deloitte.com/us/en/pages/about-deloitte/articles/about-deloitte.html>. (2014), URL: <https://www2.deloitte.com/us/en/pages/about-deloitte/articles/about-deloitte.html> (acedido em 24/10/2023).
- [10] Deloitte. «Deloitte reports record FY2023 revenue». <https://web.archive.org/web/20240312074201/https://www.deloitte.com/global/en/about/press-room/global-revenue-announcement.html>. (2022), URL: <https://www.deloitte.com/global/en/about/press-room/global-revenue-announcement.html> (acedido em 24/03/2024).
- [11] Oracle. «PL/SQL for developers». <https://web.archive.org/web/20231113002710/https://www.oracle.com/database/technologies/appdev/plsql.html>. (2019), URL: <https://www.oracle.com/database/technologies/appdev/plsql.html> (acedido em 13/11/2023).
- [12] Oracle. «SQL Language Reference». <http://archive.today/WQwCV>. (2020), URL: <https://docs.oracle.com/en/database/oracle/oracle-database/19/sqlrf/SQL-Standards.html> (acedido em 13/11/2023).
- [13] khanacademy. «Welcome to SQL». <https://web.archive.org/web/20231115235836/https://www.khanacademy.org/computing/computer-programming/sql/sql-basics/v/welcome-to-sql>. (2014), URL: <https://www.khanacademy.org/computing/computer-programming/sql/sql-basics/v/welcome-to-sql> (acedido em 15/11/2023).
- [14] Coursera. «What is exactly SQL». <https://web.archive.org/web/20231115235917/https://www.coursera.org/articles/sql-vs-mysql>. (2022), URL: <https://www.coursera.org/articles/sql-vs-mysql> (acedido em 15/11/2023).
- [15] W. Cunningham. «Principles behind the Agile Manifesto». <https://web.archive.org/web/20231119215014/https://agilemanifesto.org/principles.html>. (2002), URL: <https://agilemanifesto.org/principles.html> (acedido em 19/11/2023).
- [16] scrum.org. «What is Scrum». <https://web.archive.org/web/20231119225333/https://www.scrum.org/resources/what-scrum-module>. (2023), URL: <https://www.scrum.org/resources/what-scrum-module> (acedido em 19/11/2023).

- [17] O. Community. «OutSystems Overview - Components and Tools». (2023), URL: <https://learn.outsystems.com/training/journeys/web-developer-62/components-and-tools/o11/1308> (acedido em 20/11/2023).
- [18] O. Foundation. «About the OWASP Foundation». <https://web.archive.org/web/20231120021515/https://owasp.org/about/>. (2023), URL: <https://owasp.org/about/> (acedido em 20/11/2023).
- [19] O. Foundation. «OWASP Top Ten». <https://web.archive.org/web/20231120023714/https://owasp.org/www-project-top-ten/>. (2020), URL: <https://owasp.org/www-project-top-ten/> (acedido em 20/11/2023).
- [20] MongoDB. «What is MongoDB». <https://web.archive.org/web/20231120025850/https://www.mongodb.com/what-is-mongodb>. (2013), URL: <https://www.mongodb.com/what-is-mongodb> (acedido em 20/11/2023).
- [21] MongoDB. «What is CMMI». <https://web.archive.org/web/20231120032958/https://www.cio.com/article/274530/process-improvement-capability-maturity-model-integration-cmmi-definition-and-solutions.html>. (2021), URL: <https://www.cio.com/article/274530/process-improvement-capability-maturity-model-integration-cmmi-definition-and-solutions.html> (acedido em 20/11/2023).
- [22] Lloyd's. «Lloyd's and the Titanic». <https://web.archive.org/web/20231229153801/https://www.lloyds.com/about-lloyds/history/catastrophes-and-claims/titanic>. (2021), URL: <https://www.lloyds.com/about-lloyds/history/catastrophes-and-claims/titanic#:~:text=Back%20on%209%20January%2C%20broker,alone%20standing%20at%20%C2%A31m>. (acedido em 29/12/2023).
- [23] Propertycasualty360. «Lloyd's and the Great Fire of London». <https://web.archive.org/web/20231228022201/https://www.propertycasualty360.com/2014/02/14/history-of-adjusting-part-3-lloyds-and-the-great-f/?slreturn=20231127212158>. (2014), URL: <https://www.propertycasualty360.com/2014/02/14/history-of-adjusting-part-3-lloyds-and-the-great-f/?slreturn=20231127192631> (acedido em 28/12/2023).
- [24] L. M. Group. «The competitive position of the London Insurance Market». <https://web.archive.org/web/20231228022932/https://lmg.london/wp-content/uploads/2019/07/London-Matters-CEO-Highlights-low-res.pdf>. (2014), URL: <https://lmg.london/wp-content/uploads/2019/07/London-Matters-CEO-Highlights-low-res.pdf> (acedido em 28/12/2023).

- [25] F. M. Facts. «How the London Insurance Market works». (2018), URL: <https://www.youtube.com/watch?v=e-sMZQB4XrQ> (acedido em 28/12/2023).
- [26] Britannica. «Rate making». <https://archive.org/details/rationing-definition-types-britannica>. (2021), URL: <https://www.britannica.com/money/topic/insurance/Rate-making> (acedido em 30/12/2023).
- [27] T. daily mail. «5 million insurance policy document». <https://web.archive.org/web/20231230194955/https://tontinecoffeehouse.com/2022/09/26/insurance-and-the-titanic/>. (2013), URL: <https://www.dailymail.co.uk/news/article-2442450/5m-insurance-policy-document-Titanic-revealed-auctioneers.html> (acedido em 30/12/2023).
- [28] T. T. C. House. «Rate making». <https://web.archive.org/web/20231230194955/https://tontinecoffeehouse.com/2022/09/26/insurance-and-the-titanic/>. (2022), URL: <https://tontinecoffeehouse.com/2022/09/26/insurance-and-the-titanic/> (acedido em 30/12/2023).
- [29] O. M. Economics. «Underwriting Explained in One Minute». (2020), URL: [https://www.youtube.com/watch?v=QmJObCXq\\_Hk](https://www.youtube.com/watch?v=QmJObCXq_Hk) (acedido em 30/12/2023).
- [30] A. C. Bock e U. Frank, «Low-Code Platform», *Business and Information Systems Engineering*, vol. 63, n.º 6, pp. 733–740, dez. de 2021. DOI: [10.1007/s12599-021-00726-8](https://doi.org/10.1007/s12599-021-00726-8). URL: <https://doi.org/10.1007/s12599-021-00726-8>.
- [31] R. Sanchis, Ó. García-Perales, F. Fraile e R. Poler, «Low-Code as Enabler of Digital Transformation in Manufacturing Industry», *Applied Sciences (Switzerland)*, vol. 10, n.º 1, jan. de 2020. DOI: [10.3390/app10010012](https://doi.org/10.3390/app10010012). URL: <https://doi.org/10.3390/app10010012>.
- [32] A. Bucaioni, A. Cicchetti e F. Ciccozzi, «Modelling in Low-Code Development: A Multivocal Systematic Review», *Softw Syst Model*, 2022. DOI: [10.1007/s10270-021-00964-0](https://doi.org/10.1007/s10270-021-00964-0). URL: <https://doi.org/10.1007/s10270-021-00964-0>.
- [33] D. Di Ruscio, D. Kolovos, J. de Lara, A. Pierantonio, M. Tisi e M. Wimmer, «Low-Code Development and Model-Driven Engineering: Two Sides of the Same Coin?», *Softw Syst Model*, vol. 21, n.º 2, pp. 437–446, abr. de 2022. DOI: [10.1007/s10270-021-00970-2](https://doi.org/10.1007/s10270-021-00970-2). URL: <https://doi.org/10.1007/s10270-021-00970-2>.
- [34] M. A. A. Alamin, S. Malakar, G. Uddin, S. Afroz, T. B. Haider e A. Iqbal, «An Empirical Study of Developer Discussions on Low-Code Software Development Challenges», pp. 46–57, mai. de 2021. DOI: [10.1109/MSR52588.2021.00018](https://doi.org/10.1109/MSR52588.2021.00018). URL: <https://doi.org/10.1109/MSR52588.2021.00018>.

- [35] K. Rokis e M. Kirikova, «Exploring Low-Code Development: A Comprehensive Literature Review», *Complex Systems Informatics and Modeling Quarterly*, vol. 36, n.º 200, pp. 68–86, 2023, ISSN: 2255-9922. DOI: [10.7250/csimg.2023-36.04](https://doi.org/10.7250/csimg.2023-36.04). URL: [https://www.researchgate.net/publication/375218739\\_Exploring\\_Low-Code\\_Development\\_A\\_Comprehensive\\_Literature\\_Review](https://www.researchgate.net/publication/375218739_Exploring_Low-Code_Development_A_Comprehensive_Literature_Review).
- [36] Z. Yan, «The Impacts of Low/No-Code Development on Digital Transformation and Software Development», 2021. URL: [https://www.researchgate.net/publication/357417399\\_The\\_Impacts\\_of\\_LowNo-Code\\_Development\\_on\\_Digital\\_Transformation\\_and\\_Software\\_Development](https://www.researchgate.net/publication/357417399_The_Impacts_of_LowNo-Code_Development_on_Digital_Transformation_and_Software_Development).
- [37] OutSystems. «The State of Application Development: Is IT Ready for Disruption?» [http://web.archive.org/web/20240309153553/https://www.outsystems.com/-/media/E0A6E7121AAD4A4C975828265B3639ED.ashx?mkt\\_tok=eyJpIjoiT1RsbU56azNNakJsWVRaaiIsInQiOiIyNlBCdGlrRnVHc1VEY2c3TwSSEUwNwtTU3FBVVE0M2gwK0xoSW0xaktSZ3dWS2t6amQxOFU2WlFCR1lwR256aUhMTHVWa0ROSnZrU2tRU1Z4cTV5RFJXb2o5Wlphc21jaFR4bXY4ZmU3U3BrTkFNmm1BZm9MWkNsRHg0YjlzayJ9](https://web.archive.org/web/20240309153553/https://www.outsystems.com/-/media/E0A6E7121AAD4A4C975828265B3639ED.ashx?mkt_tok=eyJpIjoiT1RsbU56azNNakJsWVRaaiIsInQiOiIyNlBCdGlrRnVHc1VEY2c3TwSSEUwNwtTU3FBVVE0M2gwK0xoSW0xaktSZ3dWS2t6amQxOFU2WlFCR1lwR256aUhMTHVWa0ROSnZrU2tRU1Z4cTV5RFJXb2o5Wlphc21jaFR4bXY4ZmU3U3BrTkFNmm1BZm9MWkNsRHg0YjlzayJ9) (2019), URL: [https://www.outsystems.com/-/media/E0A6E7121AAD4A4C975828265B3639ED.ashx?mkt\\_tok=eyJpIjoiT1RsbU56azNNakJsWVRaaiIsInQiOiIyNlBCdGlrRnVHc1VEY2c3TwSSEUwNwtTU3FBVVE0M2gwK0xoSW0xaktSZ3dWS2t6amQxOFU2WlFCR1lwR256aUhMTHVWa0ROSnZrU2tRU1Z4cTV5RFJXb2o5Wlphc21jaFR4bXY4ZmU3U3BrTkFNmm1BZm9MWkNsRHg0YjlzayJ9](https://www.outsystems.com/-/media/E0A6E7121AAD4A4C975828265B3639ED.ashx?mkt_tok=eyJpIjoiT1RsbU56azNNakJsWVRaaiIsInQiOiIyNlBCdGlrRnVHc1VEY2c3TwSSEUwNwtTU3FBVVE0M2gwK0xoSW0xaktSZ3dWS2t6amQxOFU2WlFCR1lwR256aUhMTHVWa0ROSnZrU2tRU1Z4cTV5RFJXb2o5Wlphc21jaFR4bXY4ZmU3U3BrTkFNmm1BZm9MWkNsRHg0YjlzayJ9) (acedido em 09/03/2024).
- [38] Microsoft. «Scale up an app in Azure App Service». <https://web.archive.org/web/20240122122746/https://learn.microsoft.com/en-us/azure/app-service/manage-scale-up>. (2022), URL: <https://learn.microsoft.com/en-us/azure/app-service/manage-scale-up> (acedido em 22/01/2024).
- [39] OutSystems. «It began with a vision». <https://web.archive.org/web/20240125154654/https://www.outsystems.com/evaluation-guide/it-began-with-a-vision/>. (2019), URL: <https://www.outsystems.com/evaluation-guide/it-began-with-a-vision/> (acedido em 25/01/2024).
- [40] L. Fernandes. «A code style guide for OutSystems». <https://web.archive.org/web/20240122222825/https://leonardo-monteiro-fernandes.medium.com/a-code-style-guide-for-outsystems-97a923084159>. (2021), URL: <https://leonardo-monteiro-fernandes.medium.com/a-code-style-guide-for-outsystems-97a923084159> (acedido em 22/01/2024).

- [41] T. Sarkar. «OustSystems vs Mendix». <https://web.archive.org/web/20240125154418/https://www.appsmith.com/blog/mendix-vs-outsyste ms>. (2023), URL: <https://www.appsmith.com/blog/mendix-vs-outsyste ms> (acedido em 25/01/2024).
- [42] Google. «Google Trends on Low Code Platforms». (2006), URL: <https://trends.google.com/trends/explore?date=today%205-y&q=outsystems,Mendix,Appian,%2Fg%2F11k3224fkp,Budibase&hl=en-GB> (acedido em 26/01/2024).
- [43] Impalaintech. «Mendix vs. OutSystems vs. Appian – Best Low Code Platform?» <https://web.archive.org/web/20240126175800/https://impalaintech.com/blog/mendix-vs-outsyste ms-vs-appian/>. (2023), URL: <https://impalaintech.com/blog/mendix-vs-outsyste ms-vs-appian/> (acedido em 26/01/2024).
- [44] Mendix. «Why was Mendix Founded». <https://web.archive.org/web/20230216131432/https://www.mendix.com/evaluation-guide/why-founded/>. (2021), URL: <https://www.mendix.com/evaluation-guide/why-founded/> (acedido em 25/01/2024).
- [45] M. Community. «Create Your Own Design Systems in Mendix». (2022), URL: [https://www.youtube.com/watch?v=tt1n\\_yBeJrI](https://www.youtube.com/watch?v=tt1n_yBeJrI) (acedido em 26/01/2024).
- [46] Zippia. «Appian Company History Timeline». <https://web.archive.org/web/20240126174717/https://www.zippia.com/appian-careers-824/history/>. (2022), URL: <https://www.zippia.com/appian-careers-824/history/> (acedido em 26/01/2024).
- [47] Google. «Google Trends on Low Code Platforms». <https://web.archive.org/web/20240126174719/https://www.greatplacetowork.com/certified-company/1100833>. (2020), URL: <https://www.greatplacetowork.com/certified-company/1100833#:~:text=Appian%20was%20founded%20on%20the,a%20low%2Dcode%20automation%20platform.> (acedido em 26/01/2024).
- [48] Pega. «low code app development». <https://web.archive.org/web/20240126162659/https://www.pegacorp.com/products/platform/low-code-app-development>. (2021), URL: <https://www.pegacorp.com/products/platform/low-code-app-development> (acedido em 26/01/2024).
- [49] Pega. «O que é gestão de casos?» <https://web.archive.org/web/20240126164524/https://www.pegacorp.com/pt-br/case-management>. (2023), URL: <https://www.pegacorp.com/pt-br/case-management> (acedido em 26/01/2024).

- [50] Senthil. «Azure Cosmos DB vs. MongoDB: Which Database is Right for You?» <https://web.archive.org/web/20240128203044/https://www.whizlabs.com/blog/azure-cosmos-db-vs-mongodb/>. (2023), URL: <https://www.whizlabs.com/blog/azure-cosmos-db-vs-mongodb/#:~:text=Choosing%20between%20Cosmos%20DB%20and,low%20latency%20and%20high%20availability>. (acedido em 28/01/2024).
- [51] ProkectPro. «AWS vs Azure-Who is the big winner in the cloud war?» <https://web.archive.org/web/20240128191143/https://www.projectpro.io/article/aws-vs-azure-who-is-the-big-winner-in-the-cloud-war/401> (acedido em 28/01/2024).
- [52] Oracle. «Oracle Cloud Infrastructure Platform Overview». <https://web.archive.org/web/20240128192053/https://www.oracle.com/a/ocom/docs/cloud/oracle-cloud-infrastructure-platform-overview-wp.pdf>. (2021), URL: <https://www.oracle.com/a/ocom/docs/cloud/oracle-cloud-infrastructure-platform-overview-wp.pdf> (acedido em 28/01/2024).
- [53] 6sense. «Market Share of Oracle Cloud Infrastructure». (2023), URL: <https://6sense.com/tech/infrastructure-as-a-service/oracle-cloud-infrastructure-market-share> (acedido em 28/01/2024).
- [54] W. L. E. Team. «20 years of Atlassian, 20 lessons learned». <https://web.archive.org/web/20240128145535/https://www.atlassian.com/blog/announcements/atlassian-founders-20-years-20-lessons>. (2022), URL: <https://www.atlassian.com/blog/announcements/atlassian-founders-20-years-20-lessons> (acedido em 27/01/2024).
- [55] A. Rudder. «ClickUp Vs. Jira». <https://web.archive.org/web/20240128180201/https://www.forbes.com/advisor/in/business/software/clickup-vs-jira/>. (2023), (acedido em 28/01/2024).
- [56] E. de Sena Ramos. «Peer programming». <https://web.archive.org/web/20240205211546/https://medium.com/swlh/peer-programming-positive-and-negative-perceptions-c03141c74ea0>. (2019), URL: <https://medium.com/swlh/peer-programming-positive-and-negative-perceptions-c03141c74ea0> (acedido em 05/02/2024).
- [57] S. Faja, «Evaluating Effectiveness of Pair Programming as a Teaching Tool in Programming Courses», *Information Systems Education Journal (ISEDJ)*, vol. 12, n.º 6, pp. 36–45,

- 2014, ISSN: 1545-679X. URL: <https://files.eric.ed.gov/fulltext/EJ140923.pdf>.
- [58] J. E. Hannay, T. Dybå, E. Arisholm e D. I. Sjøberg, «The effectiveness of pair programming: A meta-analysis», *Information and Software Technology*, vol. 51, n.º 12, pp. 1110–1122, 2009, ISSN: 0950-5849. DOI: [10.1016/j.infsof.2009.02.001](https://doi.org/10.1016/j.infsof.2009.02.001). URL: [https://www.researchgate.net/publication/222408325\\_The\\_effectiveness\\_of\\_pair\\_programming\\_A\\_meta-analysis](https://www.researchgate.net/publication/222408325_The_effectiveness_of_pair_programming_A_meta-analysis).
- [59] O. Community. «OutSystems Blocks». (2023), URL: [https://learn.outsystems.com/training/journeys\(blocks-and-events-635\)/blocks/odc/57](https://learn.outsystems.com/training/journeys(blocks-and-events-635)/blocks/odc/57) (acedido em 06/02/2024).
- [60] O. Community. «OutSystems Blocks». (2023), URL: [https://learn.outsystems.com/training/journeys\(blocks-and-events-635\)/block-events/o11/78](https://learn.outsystems.com/training/journeys(blocks-and-events-635)/block-events/o11/78) (acedido em 06/02/2024).

# A Proposta de Estágio



Departamento de Engenharia Informática e de Sistemas  
Licenciaturas em Engenharia Informática

## Proposta de Estágio

Ano Letivo de 2023/2024  
1º Semestre

### Desenvolvimentos Aplicacionais

#### SUMÁRIO

Proposta de estágio para a participação num projeto de desenvolvimento aplicacional num cliente da Deloitte (nacional ou internacional).

O candidato será inserido numa equipa multidisciplinar com colaboradores com experiência nas temáticas necessárias para o desenvolvimento do projeto, nomeadamente: competências técnicas, competências de negócio, conhecimentos de metodologias de trabalho da Deloitte e *standard* de mercado, conhecimentos de gestão de projeto e valências sociais para gerir o cliente.

#### ESTAGIÁRIO

(indicar o destinatário da proposta se já estiver definido)

Número de aluno: 2020144466

Nome completo: João Filipe Silva de Almeida

#### RAMO

(indicar o(s) ramo(s) em que se enquadra)

- Desenvolvimento de Aplicações
- Redes e Administração de Sistemas
- Sistemas de Informação

#### ENTREVISTA/PROCESSO DE SELEÇÃO

(informar se o candidato indicado pelo DEIS-ISEC será submetido a uma entrevista ou outro tipo de processo de seleção antes da sua admissão efetiva. Neste ponto deverá ser tida em consideração a natureza das atividades a desenvolver, engradadas no âmbito de um projeto ou estágio curricular)

- Não
- Talvez. Especificar:
- Sim. Especificar:

## 1. ÂMBITO

Os candidatos vão ser inseridos em projetos de clientes da Deloitte com problemas reais que decorrem das atividades/tarefas dos projetos. Os candidatos serão desafiados a aprender as metodologias de trabalho da Deloitte e a trabalhar em equipas de projeto multidisciplinares. Adicionalmente, em função do projeto específico poderá ter o desafio adicional de ter de aprender uma nova tecnologia.

## 2. OBJETIVOS

O estágio proposto tem como principais funções a integração numa equipa de projeto de desenvolvimento, manutenção corretiva e evolutiva de um cliente da Deloitte.

O candidato será integrado numa equipa de projeto multidisciplinar e será acompanhado por elementos mais seniores que garantirão a supervisão necessária para cumprir os seus objetivos com sucesso.

Haverá um período de adaptação com duração variável a acordar em função da complexidade do projeto e das tecnologias envolvidas e do nível de proficiência demonstrado pelo candidato. Vindo este período de adaptação, irá integrar a equipa de projeto, tendo como principal objetivo o desenvolvimento de soluções técnicas acordadas com o cliente.

O trabalho diário será tipicamente acompanhado por um recurso com experiência nas tarefas/atividades em causa garantindo desta forma o devido acompanhamento, supervisão e revisão de todo o trabalho concretizado.

O presente projeto/estágio pretende atingir os seguintes objetivos genéricos:

- Reportar adequadamente todos os potenciais riscos para o cumprimento da tarefa que tem em mãos;
- Fazer follow-up atempado junto de outros elementos da Deloitte de todas as tarefas que estão sob a sua responsabilidade, contribuindo para a sua rápida resolução;
- Implementar de acordo com as normas de desenvolvimento e standards definidos para a iniciativa, produzindo documentação de suporte com qualidade adequada;
- Efetuar testes unitários e de sistema rigorosos e exaustivos, usando as metodologias definidas e assegurando a qualidade nas soluções implementadas;
- Adquirir conhecimentos técnicos e funcionais dos vários módulos e de como funciona a sua integração com outros sistemas;
- Comunicar de forma clara e concisa com todos os elementos da equipa, expondo as suas ideias de forma coerente. Comunicação escrita e oral em Português (intra-equipa) e inglês (se aplicável);
- Assimilar os objetivos das tarefas que lhe são pedidas e transpô-los de forma adequada para a componente técnica, pedindo ajuda a elementos mais experientes quando necessário, mas não desistindo à primeira dificuldade;
- Entregar os outputs previstos na iniciativa de acordo com os *milestones* e timings definidos no plano de entregas.

### **3. PROGRAMA DE TRABALHOS**

O projeto/estágio consistirá nas seguintes atividades e respetivas tarefas:  
(a adaptar em função do projeto/estágio)

As tarefas específicas do estágio serão definidas em função do projeto específico onde o candidato ficar alocado. Sendo que as mesmas serão definidas e acordadas por todas as partes envolvidas (candidato, coordenador ISEC e coordenador Deloitte).

### **4. CALENDARIZAÇÃO DAS TAREFAS**

O plano de trabalho deverá ser ajustado em função da data de início do estágio que deverá ser acordada com o candidato (data inicial prevista 18 de setembro). Duração: 4 meses. Após o aluno fique alocado a um projeto, este calendário será revisto.

### **5. LOCAL, HORÁRIO DE TRABALHO E CONDIÇÕES OFERECIDAS**

O local de trabalho é o edifício da Deloitte Digital Studio, sito na Rua Manuel Bandeira nº 43, 4150-479 Porto, Portugal. O horário de trabalho está compreendido entre as 9h e as 18h durante os dias úteis. Poderá haver semanas que decorrerá por sistema de híbrido com teletrabalho, cumprindo igualmente o horário estabelecido entre as 9h e 18h durante os dias úteis. O teletrabalho será acordado entre o candidato e a Deloitte por comum acordo.

### **6. TECNOLOGIAS ENVOLVIDAS**

As tecnologias envolvidas nos diversos projetos são: Java, React, Outsystems, Guidewire, TIA, Oracle, Sql Server, Service Now, SAP DS, UI Path, Katalon, AWS, Mulesoft, Microserviços, tecnologias de DevOps, tecnologias de Big Data, entre outras.

### **7. METODOLOGIA**

No decurso do estágio o aluno poderá utilizar diversas metodologias e processos tais como: Agile (preferencial), SCRUM, CMMI, ITIL, PMP, Kanban, Test Automation, DevOps, EVD (metodologia Deloitte), entre outras. As tecnologias e metodologias a utilizar no decurso do estágio dependem das necessidades específicas do projeto. O relatório de estágio deverá focar e detalhar as tecnologias e metodologias utilizadas.

## 8. ORIENTAÇÃO

*Entidade de acolhimento:*

João Escária <[jescaria@deloitte.pt](mailto:jescaria@deloitte.pt)>

*Associate Partner da Deloitte*

*DEIS-ISEC:*

*(caso já esteja definido)*

Filipe Alexandre Almeida Ningre de Sá <[filipe.sa@isec.pt](mailto:filipe.sa@isec.pt)>

*Categoria Profissional*

## B Mapa de Gantt das Formações do Estágio

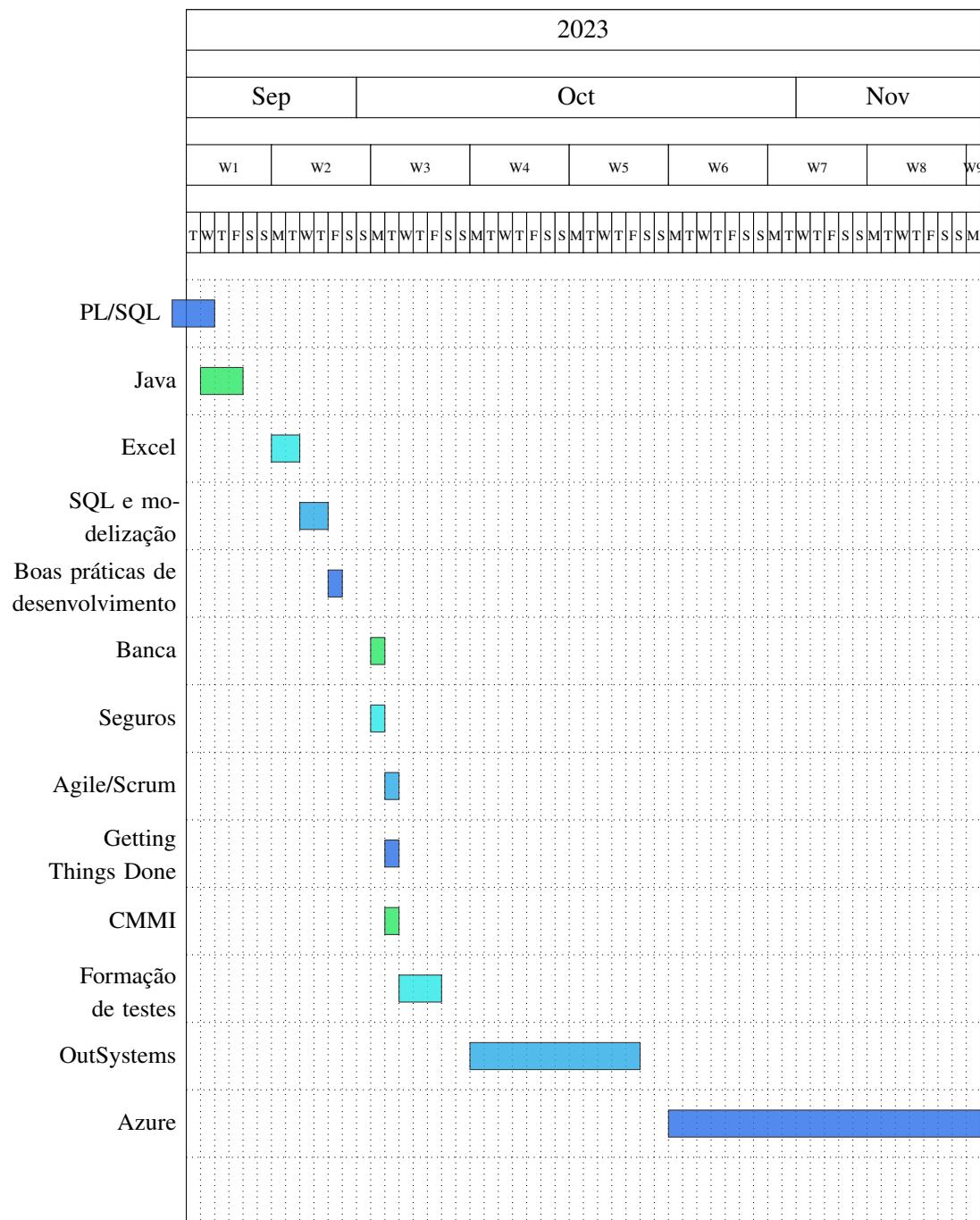


Figura 34: Mapa de Gantt das formações



## C Requisitos Funcionais dos Scripts de Python

Tabela 18: Requisito funcional *Análise de Utilizadores Inativos em GenerateMRCEmailProcess*

Nome	<b>REQ-02 — Análise de Utilizadores Inativos em GenerateMRCEmailProcess</b>
Resumo	Os utilizadores devem conseguir analisar processos GenerateMRCEmailProcess num ficheiro Excel, identificando e ignorando automaticamente no Service Center aqueles com utilizadores inativos na plataforma Atlas.
Descrição	O sistema deve permitir a análise de processos GenerateMRCEmailProcess presentes num ficheiro Excel. Utilizando a plataforma Atlas, deve identificar os processos com utilizadores inativos e ignorá-los automaticamente na plataforma Service Center.
Requisitos Relacionados	—

Tabela 19: Requisito funcional *Interrupção Controlada do Script*

Nome	<b>REQ-04 — Interrupção Controlada do Script</b>
Resumo	Os utilizadores devem conseguir interromper controladamente a execução do script em qualquer momento, utilizando a combinação de teclas CTRL + ALT + S.
Descrição	O sistema deve possibilitar ao utilizador interromper a execução do script imediatamente em qualquer ponto, utilizando a combinação de teclas CTRL + ALT + S.
Requisitos Relacionados	—

Tabela 20: Requisito funcional *Análise de Documentos em UploadDocuments\_V4*

Nome	<b>REQ-03 — Análise de Documentos em UploadDocuments_V4</b>
Resumo	Os utilizadores devem conseguir analisar processos UploadDocuments_V4 num ficheiro Excel, verificando a existência de documentos associados ao "ContextId" na base de dados do MongoDB e registando informações no Excel.
Descrição	O sistema deve permitir a análise dos processos UploadDocuments_V4 num ficheiro Excel. Deve verificar se o "ContextId" possui documentos na base de dados de Produção do MongoDB. Em caso negativo, deve assinalar no Excel a mensagem "Contract does not exist in Mongo" e prosseguir. Se existirem documentos com URIs, deve assinalar no Excel com o comentário "Document in Mongo".
Requisitos Relacionados	—

Tabela 21: Requisito funcional *Pintar Colunas com Erros de Roxo para Análise Manual*

Nome	<b>REQ-05 — Pintar Colunas com Erros de Roxo para Análise Manual</b>
Resumo	Os utilizadores devem conseguir identificar visualmente colunas com erros, pintadas a roxo no Excel, para posterior análise manual.
Descrição	O sistema deve pintar de roxo as colunas no Excel que apresentem erros, facilitando a identificação para análise manual. O script deve continuar a execução após esta identificação.
Requisitos Relacionados	REQ-01, REQ-02, REQ-03

## D Scripts de Python

### D.1 sdcprocesses-ActiveLoops.py

---

```
1  from urllib.parse import urljoin
2  import json
3  import pyautogui
4  import pyperclip
5  from selenium.webdriver.common.action_chains import
6      ↪ ActionChains
7  from selenium import webdriver
8  from selenium.webdriver.common.by import By
9  from selenium.webdriver.common.keys import Keys
10 from bs4 import BeautifulSoup
11 import re
12 import time
13 from selenium.webdriver.support.ui import WebDriverWait
14 from selenium.webdriver.support import expected_conditions as
15     ↪ EC
16 import pandas as pd
17 from selenium.common.exceptions import WebDriverException
18 from selenium.common.exceptions import WebDriverException,
19     ↪ StaleElementReferenceException
20 from selenium.webdriver.common.alert import Alert
21 import openpyxl
22 from openpyxl import load_workbook
23 from openpyxl.styles import PatternFill
24 import subprocess
25 import keyboard
26 import toml
27
28 # pip install -r requirements.txt
29
30 running = True
31
32 def stop_script():
33     global running
34     print("Script stopping on next iteration")
35     running = False
36
37 def install_dependencies():
```

```

35     with open('requirements.txt', 'r') as file:
36         dependencies = file.read().splitlines()
37
38     for dependency in dependencies:
39         subprocess.run(['pip', 'install', dependency])
40
41     def find_element(driver, xpath, timeout=10, max_attempts=5):
42         button = None
43
44         try:
45             button = WebDriverWait(driver, timeout).until(EC.
46                 element_to_be_clickable((By.XPATH, xpath)))
47         except (WebDriverException, StaleElementReferenceException)
48             as e:
49                 print(f"Catching WebDriverException: {e}")
50                 button = None
51
52         attempts = 0
53         while attempts < max_attempts and button is None:
54             try:
55                 button = WebDriverWait(driver, 1).until(EC.
56                     visibility_of_element_located((By.XPATH, xpath)))
57             except (WebDriverException,
58                     StaleElementReferenceException):
59                 attempts += 1
60                 pass
61             else:
62                 if attempts == max_attempts and button is None:
63                     raise Exception(f"Failed after {max_attempts} attempts
64                         ")
65
66         return button
67
68     def initialize_driver():
69         global driver
70         options = webdriver.ChromeOptions()
71         options.add_argument("--start-maximized")
72
73         driver = webdriver.Chrome(options) # options
74
75         with open('credentials.toml', 'r') as file:
76             credentials = toml.load(file) ['credentials']

```

```

70
71     process_id = "4850421"
72     first_url = f"https://www.ril.com/ServiceCenter/
73         ↪ Process_Instance_Detail.aspx?ProcessId={process_id}"
74     driver.get(first_url)
75
76     second_url = "https://www.ril.com/ServiceCenter/Error_Logs.
77         ↪ aspx"
78
79     # login into service center
80     driver.switch_to.window(driver.window_handles[0])
81     driver.implicitly_wait(3)
82
83     # Find the username and password fields and fill them
84     username_field = driver.find_element(By.ID, "
85         ↪ wt89_wtContentRight_wtInput1")
86     password_field = driver.find_element(By.ID, "
87         ↪ wt89_wtContentRight_wtInputPass1")
88
89     username_field.send_keys(credentials['username'])
90     password_field.send_keys(credentials['password'])
91
92     # Find and click the login button
93     login_button = driver.find_element(By.ID, "
94         ↪ wt89_wtContentRight_wt59_wtColumnsItems_wt33_wtContent_wtButton1
95         ↪ ")
96     login_button.click()
97
98     driver.execute_script(f"window.open('{second_url}', '_blank
99         ↪ ');")
100
101     def get_error_and_url(dates):
102         paint_this_row = False
103         driver.switch_to.window(driver.window_handles[1])
104         error = ""
105         url = ""
106
107         input_date_begin = find_element(driver, "/html/body/form/div
108             ↪ [3]/div[2]/div[1]/div[2]/div[2]/div[2]/div/div[1]/div/
109             ↪ input")
110         input_date_begin.send_keys(Keys.CONTROL, 'a')
111         input_date_begin.send_keys(Keys.DELETE)

```

```

102     input_date_begin.send_keys(dates[1])
103     input_date_end = find_element(driver, "/html/body/form/div"
104         ↪ [3]/div[2]/div[1]/div[2]/div[2]/div/div[2]/div/div[2]/div/
105         ↪ input")
106     input_date_end.send_keys(Keys.CONTROL, 'a')
107     input_date_end.send_keys(Keys.DELETE)
108     input_date_end.send_keys(dates[0])
109
110
111     # dropdown
112     dropdown = find_element(driver, "/html/body/form/div[3]/div"
113         ↪ [2]/div[1]/div[2]/div[1]/div/div[2]/div/div")
114     dropdown.click()
115
116     dropdown_search = find_element(driver, "/html/body/form/div"
117         ↪ [3]/div[2]/div[1]/div[2]/div[2]/div[1]/div/div[2]/div/
118         ↪ div/div[1]/div[2]/div[1]/input")
119     dropdown_search.send_keys("SDC_CS")
120     dropdown_search.send_keys(Keys.ENTER)
121
122     filter_btn = find_element(driver, "/html/body/form/div[3]/"
123         ↪ div[2]/div[1]/div[2]/div[2]/div[2]/div/div[4]/div/div/
124         ↪ input[1]")
125     filter_btn.click()
126
127     time.sleep(2) # wait for it to load
128
129     try:
130         table = find_element(driver, "/html/body/form/div[3]/div"
131             ↪ [2]/div[1]/div[2]/span[1]/div[2]/table/tbody")
132     except Exception as e:
133         print(f"Exception: Couldn't find any errors? Painting row"
134             ↪ to view later")
135         url = ""
136         error = ""
137         paint_this_row = True
138     else:
139         html_content = table.get_attribute('innerHTML')
140         soup = BeautifulSoup(html_content, 'html.parser')
141         tr_elements = soup.find_all('tr')
142         for tr in tr_elements:
143             td_elements = tr.find_all('td')

```

```

134     message = td_elements[3].get_text(strip=True)
135     if "Could not execute a built-in/extended action on
136         ↪ activity 'IsSuspended' in message:
137             continue
138     else:
139         error = td_elements[3].get_text(strip=True)
140         # get the url
141         for detail in td_elements:
142             if detail.find_all('a') and "Detail" in detail.
143                 ↪ get_text(strip=True):
144                 anchor = detail.find('a')
145                 if anchor and 'href' in anchor.attrs:
146                     relative_url = anchor['href']
147                     base_url = "https://www.ril.com/
148                         ↪ ServiceCenter/"
149                     url = urljoin(base_url, relative_url)
150                     # Now you can use the 'url' variable as
151                         ↪ needed in your code
152             else:
153                 raise Exception("Anchor element not found
154                         ↪ within detail element.")
155
156             break
157
158         if "Message: 'Leader " in error:
159             error = "Leader Error"
160         elif "DUNS provided is not valid" in error:
161             error = "DUNS Number error"
162         else:
163             paint_this_row = True
164
165     return error, url, paint_this_row
166
167
168     def get_process_last_error_window(process_id):
169         driver.switch_to.window(driver.window_handles[0])
170         url = f"https://www.ril.com/ServiceCenter/
171             ↪ Process_Instance_Detail.aspx?ProcessId={process_id}"
172         driver.get(url)
173         driver.implicitly_wait(5)
174         table = find_element(driver, "/html/body/form/div[3]/div[2]/
175             ↪ div[1]/div[2]/div[3]/table/tbody")
176         html_content = table.get_attribute('innerHTML')
177         soup = BeautifulSoup(html_content, 'html.parser')

```

```

168     tr_elements = soup.find_all('tr')
169     dates_list = []
170     # Iterate over the first two <tr> elements
171     for tr in tr_elements[:2]:
172         td_elements = tr.find_all('td')
173         if len(td_elements) >= 7:
174             date_text = td_elements[6].get_text(strip=True)
175
176             dates_list.append(date_text)
177     return dates_list
178
179 def fill_excel_and_change_to_suspended(sheet, row_index, error,
180     ↪ url, paint):
181     # Update the specified columns in the row using openpyxl
182     if not paint:
183         sheet.cell(row=row_index + 2, column=7, value="Suspended"
184             ↪ ) # column for Comment
185         sheet.cell(row=row_index + 2, column=11, value=error) #
186             ↪ Third column for PlacementId
187         sheet.cell(row=row_index + 2, column=12, value=url) # Third
188             ↪ column for PlacementId
189
190     # Check if the 'paint' variable is True, and if so, change
191         ↪ the row color
192     if paint:
193         fill = PatternFill(start_color="d69cff", end_color=""
194             ↪ d69cff", fill_type="solid") # 800080 is purple
195         for cell in sheet[row_index + 2]:
196             cell.fill = fill
197
198     if __name__ == "__main__":
199         install_dependencies()
200         # Load the Excel file with openpyxl
201         excel_file = 'SDC_processes.xlsx'
202         wb = load_workbook(excel_file)
203         sheet = wb['SDC_Generation']
204
205         keyboard.add_hotkey('ctrl+alt+s', stop_script) # stop with
206             ↪ CTRL + ALT + DELETE
207
208         initialize_driver()

```

```

202     try:
203         # analyse ProcessIds in the first column from top to
204         # ↪ bottom if the second column is empty
205         for index, row in enumerate(sheet.iter_rows(min_row=2,
206                                         # ↪ max_col=sheet.max_column, values_only=True)):
207             if (index + 2 > 1600):
208                 # only analysing from the line 1600 onwards
209                 process_id = row[0]
210                 if process_id is None: # we reached the end
211                     print("No more processIds to analyse")
212                     break
213
214             if running == False:
215                 print("script stopping...")
216                 break
217             comment_column_value = row[6]
218
219             if comment_column_value == "Active - Loop":
220                 dates = get_process_last_error_window(process_id
221                                             # ↪ )
222                 error, url, paint_this_row = get_error_and_url(
223                                             # ↪ dates)
224                 fill_excel_and_change_to_suspended(sheet, index,
225                                             # ↪ error, url, paint_this_row)
226
227                 print(f"{index+2}: {process_id}, newer date: {
228                     # ↪ dates[0]}, older date: {dates[1]}, error:
229                     # ↪ {error}, url: {url}")
230
231             finally:
232                 wb.save(excel_file)
233                 print("Excel saved")
234                 keyboard.remove_hotkey('ctrl+alt+s')
235                 driver.quit()

```

---

## D.2 mrcprocesses.py

---

```

1      import json
2      import pyautogui
3      import pyperclip
4      from selenium.webdriver.common.action_chains import
4          ActionChains

```

```
5  from selenium import webdriver
6  from selenium.webdriver.common.by import By
7  from selenium.webdriver.common.keys import Keys
8  from bs4 import BeautifulSoup
9  import re
10 import time
11 from selenium.webdriver.support.ui import WebDriverWait
12 from selenium.webdriver.support import expected_conditions as
13     EC
14 import pandas as pd
15 from selenium.common.exceptions import WebDriverException
16 from selenium.common.exceptions import WebDriverException,
17     StaleElementReferenceException
18 from selenium.webdriver.common.alert import Alert
19 import openpyxl
20 from openpyxl import load_workbook
21 import subprocess
22 import keyboard
23 import toml
24
25 # pip install -r requirements.txt
26
27 running = True
28
29 def stop_script():
30     global running
31     print("Script stopping on next iteration")
32     running = False
33
34 def install_dependencies():
35     with open('requirements.txt', 'r') as file:
36         dependencies = file.read().splitlines()
37
38         for dependency in dependencies:
39             subprocess.run(['pip', 'install', dependency])
40
41 def find_element(driver, xpath, timeout=10, max_attempts=20):
42     button = None
43
44     try:
45         button = WebDriverWait(driver, timeout).until(EC.
46             element_to_be_clickable((By.XPATH, xpath)))
47     except:
48         max_attempts -= 1
49
50     if max_attempts < 0:
51         raise WebDriverException("Element not found after "
52             "multiple attempts")
53
54     return button
```



```

76     # login into service center
77     driver.switch_to.window(driver.window_handles[0])
78     driver.implicitly_wait(3)
79
80     # Find the username and password fields and fill them
81     username_field = driver.find_element(By.ID, "
82         ↪ wt89_wtContentRight_wtInput1")
83     password_field = driver.find_element(By.ID, "
84         ↪ wt89_wtContentRight_wtInputPass1")
85
86     username_field.send_keys(credentials['username'])
87     password_field.send_keys(credentials['password'])
88
89     # Find and click the login button
90     login_button = driver.find_element(By.ID, "
91         ↪ wt89_wtContentRight_wt59_wtColumnsItems_wt33_wtContent_wtButton1
92         ↪ ")
93     login_button.click()
94
95
96     el = WebDriverWait(driver, 30).until(EC.
97         ↪ visibility_of_element_located((By.XPATH, "//button[
98             ↪ @type='submit' and contains(., 'Next')]))))
99
100    WebDriverWait(driver, 10).until(lambda d: 'false' in el.
101        ↪ get_attribute('aria-disabled'))
102    el.click()
103
104    try:
105        WebDriverWait(driver, 200).until(EC.
106            ↪ visibility_of_element_located((By.XPATH, "/html/
107            ↪ body/div[1]/div/div/div[3]/div[1]/div[3]/main/div/
108            ↪ div/div[2]/div[2]/div[2]/div[5]/div/div[1]/div/form
109            ↪ /div/div[2]/div/div/div/div/div/div[2]/div[1]")
110            ↪ ))
111    except (WebDriverException, StaleElementReferenceException)
112        ↪ as e:
113        print(f"Catching WebDriverException: {e}")

```

```

104     mongoloaded = None
105     while mongoloaded is None:
106         try:
107             mongoloaded = WebDriverWait(driver, 1).until(EC.
108                 visibility_of_element_located((By.XPATH, "/
109                 ↪ html/body/div[1]/div/div/div[3]/div[1]/div
110                 ↪ [3]/main/div/div/div[2]/div[2]/div[2]/div[5]/
111                 ↪ div/div[1]/div/form/div/div[2]/div/div/div/
112                 ↪ div/div/div[2]/div[1]")))
113             except (WebDriverException,
114                 ↪ StaleElementReferenceException):
115                 pass
116
117     def get_process_info(process_id):
118         # URL for the service center processes
119         url = f"https://www.ril.com/servicecenter/
120             ↪ Process_Instance_Detail.aspx?ProcessId={process_id}"
121
122         driver.switch_to.window(driver.window_handles[0])
123         driver.get(url)
124
125         driver.implicitly_wait(5)
126
127         target_div_id = "wt70_wtContentTop_wtListRecords1"
128         target_div = driver.find_element(By.ID, target_div_id)
129
130         html_content = target_div.get_attribute('innerHTML')
131
132         soup = BeautifulSoup(html_content, 'html.parser')
133
134         result = {}
135
136         # Find all div elements with a specific class
137         top_level_divs = soup.find_all('div', class_='columns gutter
138             ↪ -base align-items-center margin-bottom-xs')
139
140         # Loop through each div element
141         for div in top_level_divs:
142             # Find the span element inside the first div
143             field_name_element = div.find('span', class_='text-
144                 ↪ neutral-7')

```

```

136
137     # Find the content inside the second div
138     second_div = div.find('div', class_='columns-item').
139         ↪ find_next('div', class_='columns-item')
140
141     # Check if both elements are present
142     if field_name_element and second_div:
143         field_name = field_name_element.get_text(strip=True)
144         field_value = second_div.get_text(strip=True)
145
146         # Add the field and its corresponding value to the
147             ↪ result dictionary
148         result[field_name] = field_value
149     else:
150         print(f"div_elements:\n{top_level_divs}")
151         print(f"div:\n{div}")
152         print(f"html_content:\n{html_content}")
153         print(f"result:\n{result}")
154         raise Exception("Error getting some of the process
155             ↪ info")
156
157     return result
158
159
160     def is_user_active(uwxid):
161         driver.switch_to.window(driver.window_handles[1])
162         time.sleep(1) # if it's too fast, it will deselected after
163             ↪ clicking
164         element = find_element(driver, "/html/body/div[1]/div/div/
165             ↪ div[3]/div[1]/div[3]/main/div/div/div[2]/div[2]/div
166             ↪ [2]/div[5]/div/div[1]/div/form/div/div[2]/div/div/div/
167             ↪ div/div/div[2]/div[1]")
168         element.click()
169         driver.implicitly_wait(1)
170         # search
171         pyautogui.hotkey('ctrl', 'a')
172         pyautogui.press('delete')
173         text_query = ' {_id: "' + uwxid + '"}'
174         pyperclip.copy(text_query)
175         pyautogui.hotkey('ctrl', 'v')
176         driver.implicitly_wait(1)
177         pyautogui.press('enter')

```

```

170
171     # Wait for loading documents to appear and disappear
172     try:
173         WebDriverWait(driver, 5
174             ).until(EC.presence_of_element_located((By.XPATH, "//
175             ↪ div[contains(@class, 'data-explorer-subheader-
176             ↪ info-is-loading') and contains(text(), 'Loading
177             ↪ Documents')]")))
178         WebDriverWait(driver, 10
179             ).until_not(EC.presence_of_element_located((By.XPATH,
180                 ↪ "//div[contains(@class, 'data-explorer-subheader-
181                 ↪ info-is-loading') and contains(text(), 'Loading
182                 ↪ Documents')]")))
183     except (WebDriverException, StaleElementReferenceException)
184         ↪ as e:
185         print(f"Catching WebDriverException: {e}")
186         element = find_element(driver, "/html/body/div[1]/div/div
187             ↪ /div[3]/div[1]/div[3]/main/div/div/div[2]/div[2]/
188             ↪ div[2]/div[5]/div/div[1]/div/form/div/div[2]/div/
189             ↪ div/div/div/div/div[2]/div[1]")
190         element.click()
191         driver.implicitly_wait(1)
192         # search
193         pyautogui.hotkey('ctrl', 'a')
194         pyautogui.press('delete')
195         text_query = ' {_id: "' + uwqid + '"}'
196         pyperclip.copy(text_query)
197         pyautogui.hotkey('ctrl', 'v')
198         driver.implicitly_wait(1)
199         pyautogui.press('enter')
200         time.sleep(5) # alternatively just wait 5 seconds
201
202         # move mouse over query result for copy button to become
203             ↪ visible
204         query_result = find_element(driver, "/html/body/div[1]/div/
205             ↪ div/div[3]/div[1]/div[3]/main/div/div/div[2]/div[2]/
206             ↪ div[2]/div[5]/div/div[3]/div/div[2]")
207         ActionChains(driver).move_to_element(query_result).perform()
208         driver.implicitly_wait(10)
209
210         # copy the query result

```

```
198     button = find_element(driver, "/html/body/div[1]/div/div/div
199         ↪ [3]/div[1]/div[3]/main/div/div[2]/div[2]/div[2]/
200         ↪ div[5]/div/div[3]/div/div[2]/div/div/div[2]/div/button
201         ↪ ")
202
203     driver.implicitly_wait(10)
204     button.click()
205     clipboard_content = pyperclip.paste()
206     data = json.loads(clipboard_content) # Parse the JSON string
207     active_value = data.get('active', None) # Extract the value
208         ↪ of the 'active' key
209
210
211     return active_value
212
213
214
215     def fill_in_excel_openpyxl(sheet, row_index, process_data):
216
217         # Extract relevant information from process_data
218         contract_id = process_data.get('ContractId', None)
219         placement_id = process_data.get('PlacementId', None)
220         section_id = process_data.get('SectionId', None)
221         submission_id = process_data.get('SubmissionId', None)
222         uwx_id = process_data.get('UWXId', None)
223
224
225         # Update the specified columns in the row using openpyxl
226         sheet.cell(row=row_index + 2, column=2, value=contract_id) #
227             ↪ Second column for ContractId
228         sheet.cell(row=row_index + 2, column=3, value=placement_id)
229             ↪ # Third column for PlacementId
230         sheet.cell(row=row_index + 2, column=4, value=section_id) #
231             ↪ Fourth column for SectionId
232         sheet.cell(row=row_index + 2, column=5, value=submission_id)
233             ↪ # Fifth column for SubmissionId
234         sheet.cell(row=row_index + 2, column=7, value=uwx_id) #
235             ↪ Seventh column for UWXId
236
237
238     def fill_excel_and_skip_process_openpyxl(sheet, row_index):
239
240         skipped = True
241         sheet.cell(row=row_index + 2, column=6, value="User not
242             ↪ active") # Seventh column for "User not active"
243
244
245         # Skip the process automatically
246         driver.switch_to.window(driver.window_handles[0])
247
248         try:
```

```

229     skip_button = WebDriverWait(driver, 5).until(EC.
230         ↪ element_to_be_clickable((By.XPATH, "/html/body/form
231         ↪ /div[3]/div[2]/div[1]/div[2]/div[3]/table/tbody/tr
232         ↪ [1]/td[11]/input[2]")))
233     driver.implicitly_wait(1)
234     skip_button.click()
235     WebDriverWait(driver, 10).until(EC.alert_is_present())
236     time.sleep(0.5)
237     driver.switch_to.alert.accept()
238     time.sleep(3)
239
240     except:
241         print("Couldn't skip, saving without 'Y' in Skipped")
242         skipped = False
243
244     if skipped:
245         sheet.cell(row=row_index + 2, column=8, value="Y") #
246             ↪ Ninth column for "Y"
247
248     return skipped
249
250
251     if __name__ == "__main__":
252         install_dependencies()
253
254         # Load the Excel file with openpyxl
255         excel_file = 'SDC_processes.xlsx'
256         wb = load_workbook(excel_file)
257         sheet = wb['GenerateMRCEMail']
258
259         keyboard.add_hotkey('ctrl+alt+s', stop_script) # stop with
260             ↪ CTRL + ALT + DELETE
261
262         initialize_driver()
263
264         try:
265             # analyse ProcessIds in the first column from top to
266             ↪ bottom if the second column is empty
267             for index, row in enumerate(sheet.iter_rows(min_row=2,
268                 ↪ max_col=sheet.max_column, values_only=True)):
269                 process_id = row[0]
270
271                 if process_id is None: # we reached the end
272                     print("No more processIds to analyse")
273                     break
274
275                 if running == False:

```

```

263         print("script stopping...")
264         break
265     second_column_value = row[1]
266
267     if second_column_value is None:
268         process_data = get_process_info(process_id)
269         uwqid = process_data.get('UWXID', None) # Extract
270             ↪ UWXID and store it in uwqid variable
271         user_active = is_user_active(uwqid)
272
273         if not user_active:
274             fill_excel_and_skip_process_openpyxl(sheet,
275                 ↪ index)
276             fill_in_excel_openpyxl(sheet, index, process_data)
277
278     print(f"{index + 2}: {'Active' if user_active else
279             ↪ 'Inactive'}, {process_id}", {process_data.get
280             ↪ ('ContractId', None)[:8]}, {process_data.get(
281             ↪ ('PlacementId', None)[:8]}, {process_data.get(
282             ↪ ('SectionId', None)[:8]}, {process_data.get(
283             ↪ ('SubmissionId', None)[:8]}, {process_data.get(
284             ↪ ('UWXID', None)[:8]})

285
286     finally:
287         wb.save(excel_file)
288         print("Excel saved")
289         keyboard.remove_hotkey('ctrl+alt+s')
290         driver.quit()

```

---

### D.3 4VDocumentsprocesses.py

```

1  import json
2  import pyautogui
3  import pyperclip
4  from selenium.webdriver.common.action_chains import
5      ↪ ActionChains
6  from selenium import webdriver
7  from selenium.webdriver.common.by import By
8  from selenium.webdriver.common.keys import Keys
9  from bs4 import BeautifulSoup
import re

```

```

10     import time
11     from selenium.webdriver.support.ui import WebDriverWait
12     from selenium.webdriver.support import expected_conditions as
13         ↪ EC
14     import pandas as pd
15     from selenium.common.exceptions import WebDriverException
16     from selenium.common.exceptions import WebDriverException,
17         ↪ StaleElementReferenceException
18     from selenium.webdriver.common.alert import Alert
19     import openpyxl
20     from openpyxl import load_workbook
21     import subprocess
22     import keyboard
23     from openpyxl.styles import PatternFill
24     import toml
25
26     # pip install -r requirements.txt
27
28     running = True
29
30     def stop_script():
31         global running
32         print("Script stopping on next iteration")
33         running = False
34
35     # Set a global hotkey (e.g., Ctrl + Alt + S) to stop the script
36     keyboard.add_hotkey('ctrl+alt+s', stop_script)
37
38     def install_dependencies():
39         with open('requirements.txt', 'r') as file:
40             dependencies = file.read().splitlines()
41
42             for dependency in dependencies:
43                 subprocess.run(['pip', 'install', dependency])
44
45     def find_element(driver, xpath, timeout=10, max_attempts=20):
46         button = None
47         try:
48             button = WebDriverWait(driver, timeout).until(EC.
49                 ↪ element_to_be_clickable((By.XPATH, xpath)))

```

```

47     except (WebDriverException, StaleElementReferenceException):
48         ↪ as e:
49             # print(f"Catching WebDriverException: {e}")
50             button = None
51
52     attempts = 0
53     while attempts < max_attempts and button is None:
54         try:
55             button = WebDriverWait(driver, 1).until(EC.
56                 ↪ visibility_of_element_located((By.XPATH, xpath))
57                 ↪ )
58         except (WebDriverException,
59                 ↪ StaleElementReferenceException):
60             attempts += 1
61             pass
62     else:
63         if attempts == max_attempts and button is None:
64             raise Exception(f"Failed after {max_attempts} attempts
65                 ↪ ")
66
67     return button
68
69
70     def initialize_driver():
71         global driver
72         options = webdriver.ChromeOptions()
73         options.add_argument("--start-maximized")
74
75         driver = webdriver.Chrome(options) # options
76
77         with open('credentials.toml', 'r') as file:
78             credentials = toml.load(file)['credentials']
79
80             process_id = "2101221"
81             first_url = f"https://www.ril.com/servicecenter/
82                 ↪ Process_Instance_Detail.aspx?ProcessId={process_id}"
83             driver.get(first_url)
84
85             second_url = "https://cloud.mongodb.com/v2/634680405
86                 ↪ a74750c3ee0221c#/metrics/replicaSet/6346878
87                 ↪ bf630861381299e43/explorer/application/documents/find"
88             driver.execute_script(f"window.open('{second_url}', '_blank
89                 ↪ ');")

```

```

79
80     # login into service center
81     driver.switch_to.window(driver.window_handles[0])
82     driver.implicitly_wait(3)
83
84     # Find the username and password fields and fill them
85     username_field = driver.find_element(By.ID, "
86         ↪ wt89_wtContentRight_wtInput1")
86     password_field = driver.find_element(By.ID, "
87         ↪ wt89_wtContentRight_wtInputPass1")
88
89     username_field.send_keys(credentials['username'])
90     password_field.send_keys(credentials['password'])
91
91     # Find and click the login button
92     login_button = driver.find_element(By.ID, "
93         ↪ wt89_wtContentRight_wt59_wtColumnsItems_wt33_wtContent_wtButton1
94         ↪ ")
95     login_button.click()
96     driver.switch_to.window(driver.window_handles[1])
97     username_field = driver.find_element(By.ID, "username")
98
99     username_field.send_keys(credentials['email'])
100
100     el = WebDriverWait(driver, 30).until(EC.
101         ↪ visibility_of_element_located((By.XPATH, "//button[
102             ↪ @type='submit' and contains(., 'Next')]")))
103     WebDriverWait(driver, 10).until(lambda d: 'false' in el.
104         ↪ get_attribute('aria-disabled'))
105     el.click()
106
106     try:
107         WebDriverWait(driver, 200).until(EC.
108             ↪ visibility_of_element_located((By.XPATH, "/html/
109                 ↪ body/div[1]/div/div/div[3]/div[1]/div[3]/main/div/
110                     ↪ div/div[2]/div[2]/div[2]/div[5]/div/div[1]/div/form
111                     ↪ /div/div[2]/div/div/div/div/div/div[2]/div[1]")
112                     ↪ )))
113     except (WebDriverException, StaleElementReferenceException)
114         ↪ as e:
115             print(f"Catching WebDriverException: {e}")

```

```

107     mongoloaded = None
108     while mongoloaded is None:
109         try:
110             mongoloaded = WebDriverWait(driver, 1).until(EC.
111                 visibility_of_element_located((By.XPATH, "
112                 ↪ html/body/div[1]/div/div/div[3]/div[1]/div
113                 ↪ [3]/main/div/div/div[2]/div[2]/div[2]/div[5]/
114                 ↪ div/div[1]/div/form/div/div[2]/div/div/div/
115                 ↪ div/div/div/div[2]/div[1]")))
116         except (WebDriverException,
117                 ↪ StaleElementReferenceException):
118             pass
119
120
121     def get_process_info(process_id):
122         # URL for the service center processes
123         url = f"https://www.ril.com/servicecenter/
124             ↪ Process_Instance_Detail.aspx?ProcessId={process_id}"
125
126         driver.switch_to.window(driver.window_handles[0])
127         driver.get(url)
128
129         driver.implicitly_wait(5)
130
131
132         target_div_id = "wt70_wtContentTop_wtListRecords1"
133         target_div = driver.find_element(By.ID, target_div_id)
134
135         html_content = target_div.get_attribute('innerHTML')
136
137         soup = BeautifulSoup(html_content, 'html.parser')
138
139         result = {}
140
141
142         # Find all div elements with a specific class
143         top_level_divs = soup.find_all('div', class_='columns gutter
144             ↪ -base align-items-center margin-bottom-xs')
145
146
147         # Loop through each div element
148         for div in top_level_divs:
149             # Find the span element inside the first div
150             field_name_element = div.find('span', class_='text-
151                 ↪ neutral-7')

```

```

139
140     # Find the content inside the second div
141     second_div = div.find('div', class_='columns-item').
142         ↪ find_next('div', class_='columns-item')
143
144     # Check if both elements are present
145     if field_name_element and second_div:
146         field_name = field_name_element.get_text(strip=True)
147         field_value = second_div.get_text(strip=True)
148
149         # Add the field and its corresponding value to the
150             ↪ result dictionary
151         result[field_name] = field_value
152
153     else:
154         print(f"div_elements:\n{top_level_divs}")
155         print(f"div:\n{div}")
156         print(f"html_content:\n{html_content}")
157         print(f"result:\n{result}")
158         raise Exception("Error getting some of the process
159             ↪ info")
160
161     return result
162
163
164     def analyse_process(process_data):
165         #contextid = process_data['ContextId']
166         context_id = process_data.get('ContextId', {})
167
168         driver.switch_to.window(driver.window_handles[1])
169         time.sleep(1) # if it's too fast, it will deselected after
170             ↪ clicking
171         element = find_element(driver, "/html/body/div[1]/div/div/
172             ↪ div[3]/div[1]/div[3]/main/div/div/div[2]/div[2]/div
173             ↪ [2]/div[5]/div/div[1]/div/form/div/div[2]/div/div/div/
174             ↪ div/div/div[2]/div[1]")
175         element.click()
176         driver.implicitly_wait(1)
177
178         # search
179         pyautogui.hotkey('ctrl', 'a')
180         pyautogui.press('delete')
181         text_query = '{"context._xid": "' + context_id + '"}'
182         pyperclip.copy(text_query)

```

```

173     pyautogui.hotkey('ctrl', 'v')
174     driver.implicitly_wait(1)
175     pyautogui.press('enter')
176
177     # Wait for loading documents to appear and disappear
178     try:
179         WebDriverWait(driver, 5
180             ).until(EC.presence_of_element_located((By.XPATH, "//
181             ↪ div[contains(@class, 'data-explorer-subheader-
182             ↪ info-is-loading') and contains(text(), 'Loading
183             ↪ Documents')]")))
184         WebDriverWait(driver, 10
185             ).until_not(EC.presence_of_element_located((By.XPATH,
186                 ↪ "//div[contains(@class, 'data-explorer-subheader-
187                 ↪ info-is-loading') and contains(text(), 'Loading
188                 ↪ Documents')]")))
189     except (WebDriverException, StaleElementReferenceException)
190         as e:
191         print(f"Catching WebDriverException: waiting 5 seconds
192             ↪ instead")
193         element = find_element(driver, "/html/body/div[1]/div/div
194             ↪ /div[3]/div[1]/div[3]/main/div/div/div[2]/div[2]/
195             ↪ div[2]/div[5]/div/div[1]/div/form/div/div[2]/div/
196             ↪ div/div/div/div/div[2]/div[1]")
197         element.click()
198         driver.implicitly_wait(1)
199         # search
200         pyautogui.hotkey('ctrl', 'a')
201         pyautogui.press('delete')
202         text_query = '{"context._xid": "' + context_id + '"}'
203         pyperclip.copy(text_query)
204         pyautogui.hotkey('ctrl', 'v')
205         driver.implicitly_wait(1)
206         pyautogui.press('enter')
207
208         time.sleep(5) # alternatively just wait 5 seconds
209
210     # move mouse over query result for copy button to become
211         ↪ visible
212     try:

```

```

201     query_result = find_element(driver, "/html/body/div[1]/
202         ↪ div/div/div[3]/div[1]/div[3]/main/div/div/div[2]/
203         ↪ div[2]/div[2]/div[5]/div/div[3]/div/div[2]",
204         ↪ timeout=4, max_attempts=0)
205     ActionChains(driver).move_to_element(query_result).
206         ↪ perform()
207     driver.implicitly_wait(10)
208     first_button = find_element(driver, "/html/body/div[1]/
209         ↪ div/div/div[3]/div[1]/div[3]/main/div/div/div[2]/
210         ↪ div[2]/div[2]/div[5]/div/div[3]/div/div[2]/div/div/
211         ↪ div[2]/div/button", timeout=4, max_attempts=0)
212     except:
213         # Couldn't find any documents
214         return "Contract does not exist in Mongo", False
215
216     list = find_element(driver, "/html/body/div[1]/div/div/div
217         ↪ [3]/div[1]/div[3]/main/div/div/div[2]/div[2]/
218         ↪ div[5]/div/div[3]")
219     div_elements = list.find_elements(By.CSS_SELECTOR, 'div.data
220         ↪ -explorer-results-card')
221
222     top_of_the_box = find_element(driver, "/html/body/div[1]/div
223         ↪ /div/div[3]/div[1]/div[3]/main/div/div/div[2]/div[2]/
224         ↪ div[2]/div[5]/div/div[3]/div/div[1]/div[2]")
225     top_of_the_box.click()
226
227     print ("uri_value: ", end='')
228     for i in range(2, 2 + len(div_elements)):
229         #copy_button = find_element(driver, f"/html/body/div[1]/
230             ↪ div/div/div[3]/div[1]/div[3]/main/div/div/div[2]/
231             ↪ div[2]/div[2]/div[5]/div/div[3]/div/div[{i}]/div/
232             ↪ div/div[2]/div/button")
233
234     copy_button = None
235     while True:
236         pyautogui.press('pagedown')
237         try:
238             query_result = find_element(driver, f"/html/body/
239                 ↪ div[1]/div/div/div[3]/div[1]/div[3]/main/div/
240                 ↪ div/div[2]/div[2]/div[2]/div[5]/div/div[3]/
241                 ↪ div/div[{i}]", timeout=5, max_attempts=0)

```

```

224
225     ActionChains(driver).move_to_element(query_result) .
226         → perform()
227     driver.implicitly_wait(10)
228
229     driver.execute_script("arguments[0].scrollIntoView(
230         → true);", query_result)
231     copy_button = find_element(driver, f"/html/body/div
232         → [1]/div/div/div[3]/div[1]/div[3]/main/div/div
233         → /div[2]/div[2]/div[2]/div[5]/div/div[3]/div/
234         → div[{i}]/div/div/div[2]/div/button", timeout
235         → =5, max_attempts=0)
236
237     driver.execute_script("arguments[0].scrollIntoView(
238         → true);", copy_button)
239     except:
240         copy_button = None
241     if copy_button != None:
242         break
243
244     ActionChains(driver).move_to_element(query_result) .
245         → perform()
246     driver.implicitly_wait(10)
247     copy_button.click()
248
249     clipboard_content = pyperclip.paste()
250     data = json.loads(clipboard_content) # Parse the JSON
251         → string
252     uri_value = data.get('uri', None) # Extract the value of
253         → the 'active' key
254     print("{", end='')
255     print(f"uri_value[:8]}", end=' ')
256     print("}", ", end=' ')
257
258     if uri_value == '':
259         return "ERROR", True
260     print("")
261     return "Document in Mongo", False
262
263 def fill_excel_and_skip_process_openpyxl(sheet, row_index,
264     → process_data, comment, paint_this_row_and_dont_skip):

```

```

254     sheet.cell(row=row_index + 2, column=2, value=process_data.
255                 ↪ get('DocumentsUploadProcessId', None))
256     sheet.cell(row=row_index + 2, column=3, value=process_data.
257                 ↪ get('ContextType', None))
258     sheet.cell(row=row_index + 2, column=4, value=process_data.
259                 ↪ get('ContextId', None))
260     sheet.cell(row=row_index + 2, column=5, value=comment)
261
262
263
264
265
266     # Check if the 'paint' variable is True, and if so, change
267     # the row color
268     if paint_this_row_and_dont_skip:
269         fill = PatternFill(start_color="d69cff", end_color="
270                           ↪ d69cff", fill_type="solid") # 800080 is the
271                           ↪ hexadecimal code for purple
272         for cell in sheet[row_index + 2]:
273             cell.fill = fill
274
275     return
276
277
278     # Skip the process automatically
279     skipped = True
280     driver.switch_to.window(driver.window_handles[0])
281     try:
282         skip_button = WebDriverWait(driver, 5).until(EC.
283             ↪ element_to_be_clickable((By.XPATH, "/html/body/form
284             ↪ /div[3]/div[2]/div[1]/div[2]/div[3]/table/tbody/tr
285             ↪ [1]/td[11]/input[2]")))
286         driver.implicitly_wait(1)
287         skip_button.click()
288         WebDriverWait(driver, 10).until(EC.alert_is_present())
289         time.sleep(0.5)
290         driver.switch_to.alert.accept()
291         time.sleep(3)
292     except:
293         print("Couldn't skip, saving without 'Y' in Skipped")
294         skipped = False
295
296
297     if skipped:
298         sheet.cell(row=row_index + 2, column=6, value="Y")
299
300
301 if __name__ == "__main__":
302     install_dependencies()

```

```

286
287     # Load the Excel file with openpyxl
288     excel_file = 'SDC_processes.xlsx'
289     wb = load_workbook(excel_file)
290     sheet = wb['Upload_Documents_V4']
291
292     keyboard.add_hotkey('ctrl+alt+s', stop_script) # stop with
293     ↪ CTRL + ALT + DELETE
294
295     initialize_driver()
296     try:
297         # analyse ProcessIds in the first column from top to
298         ↪ bottom if the second column is empty
299         for index, row in enumerate(sheet.iter_rows(min_row=2,
300                                         ↪ max_col=sheet.max_column, values_only=True)):
301             process_id = row[0]
302             if process_id is None: # we reached the end
303                 print("No more processIds to analyse")
304                 break
305             if running == False:
306                 print("script stopping...")
307                 break
308             second_column_value = row[1]
309
310             if second_column_value is None:
311                 process_data = get_process_info(process_id)
312                 comment, paint_this_row_and_dont_skip =
313                 ↪ analyse_process(process_data)
314                 fill_excel_and_skip_process_openpyxl(sheet, index,
315                 ↪ process_data, comment,
316                 ↪ paint_this_row_and_dont_skip)
317
318                 print(f"{index + 2}: {process_id}, {comment}, {
319                 ↪ paint_this_row_and_dont_skip}, {process_data.
320                 ↪ get('ContextId', None) [:8]}, {process_data.
321                 ↪ get('ContextType', None)}, {process_data.get('
322                 ↪ DocumentsUploadProcessId', None) [:8]}")
323
324     finally:
325         wb.save(excel_file)
326         print("Excel saved")

```

```
317     keyboard.remove_hotkey('ctrl+alt+s')
318     driver.quit()
```

---

## D.4 requirements.txt

---

```
1 json
2 pyautogui
3 pyperclip
4 selenium
5 beautifulsoup4
6 pandas
7 openpyxl
8 keyboard
9 toml
```

---

---

Este relatório foi gerado utilizando LaTeX. Os arquivos-fonte estão disponíveis em <https://github.com/Yeshey/DeloitteInternshipReport-Latex>.