

Otimização de Plataformas Informáticas no Setor Bancário

João Paulo Baptista Pereira - a36769

*Relatório de Estágio apresentado à Escola Superior de Tecnologia e Gestão para
obtenção do Grau de Mestre em Informática*

Trabalho realizado sob a orientação de:

Professor Doutor José Luís Padrão Exposto (IPB)

Eng. Manuel Pinto Teixeira da Silva (ITSector)

Bragança

Maio 2022

Agradecimentos

Quero agradecer a todos os docentes e colegas que contribuíram com o seu conhecimento e sinergias para o meu desenvolvimento ao longo deste percurso académico.

Gostaria de agradecer particularmente ao orientador da dissertação, o Professor Doutor José Exposto, por toda a sua disponibilidade e pelas suas excelentes propostas ao longo do meu percurso académico.

Um grande agradecimento também para a ITSector, por ter possibilitado a realização do estágio.

Por fim um agradecimento em especial ao meu Pai, á minha namorada, aos meus restantes familiares e a todos os meus amigos que me acompanharam com o seu apoio ao longo deste ano.

Resumo

A importância do acesso a informação de qualidade em tempo real é um fator crítico de sucesso em qualquer setor de atividade. No caso do setor bancário, a concorrência é extremamente agressiva, levando os bancos a recorrer a plataformas informáticas capazes de integrar todo o processo organizacional e permitir uma melhor a ligação/iteração aos seus clientes. Os bancos estão a adaptar-se a esta revolução digital, o que implica levar a cabo uma alteração significativa nos modelos de negócio e na cultura interna das organizações. Para conseguir sobreviver, a banca tradicional está a adaptar-se através da simplificação e digitalização dos seus processos, tornando-os menos burocráticos e desta forma conseguir reduzir custos (redução do número de balcões físicos).

A utilização de TI permite ao setor bancário, não só uma redução de custos, através da simplificação dos seus processos operacionais, e redução de recursos humanos, como também uma maior adaptação aos clientes dos dias de hoje e melhorar a sua relação com eles com recurso a técnicas de *data-mining* e AI. Os serviços *E-Banking* e *M-Banking* permitem-nos realizar um conjunto variado de operações bancárias através da internet e dos dispositivos móveis e que têm vindo a substituir os serviços tradicionais. Desta forma, a utilização de plataformas à medida do cliente, torna as empresas de TI extremamente importantes neste contexto atual.

O presente relatório descreve o trabalho realizado, ao longo do estágio profissional realizado no âmbito do segundo ano do mestrado de Informática, na empresa ITSector (01/09/2021 a 31/03/2022), na área de suporte a plataformas informáticas do setor bancário. Na primeira parte do estágio foi necessário proceder à instalação e configuração dos cinco projetos internacionais atribuídos. Numa segunda parte, foram atribuídos tickets de baixa/media complexidade, onde já foi necessário reunir com os clientes, identificar requisitos, desenhar a solução, e implementar. Na última parte do estágio, foi atribuída a responsabilidade pela manutenção/otimização de um projeto internacional, onde foi necessário desenvolver novos módulos, alterar código existente (correções evolutivas personalizadas que surgiram por parte do cliente), e otimizar bases de dados.

Palavras-Chave: Tecnologias de Informação, Sistema de informação, *E-Banking*, *M-Banking*, Setor bancário.

Abstract

The importance of to quality data in real time is a critical success factor in any sector of activity. In the case of the banking sector, this competition is extremely aggressive, leading banks to choose IT platforms capable of integrating the entire organizational process and allowing better connection/iteration with their customers. Banks are adapting to this digital revolution, which implies carrying out a significant change in business models and in the internal culture of organizations. To survive, traditional banking is adapting through the simplification and digitization of its processes, making them less bureaucratic and consequently reducing costs (reduction the number of physical stores).

The use of IT seeks not only to reduce costs, through the simplification of its operational processes, and reduction of human resources, but also a greater adaptation to today's customers and improve its relationship with the banks using techniques of data mining and AI. E-Banking and M-Banking services allowing us to carry out a wide range of banking operations via the internet and mobile devices, which have been replacing traditional services. In this way, the implementation and maintenance of customized platforms makes IT companies extremely important in this current context.

This report describes the work carried out, during the internship carried out within the scope of the second year of the Master's in Informatics, at the company ITSector (09/01/2021 to 03/31/2022), in support area for IT platforms in the banking sector. In the first part of the internship, it was necessary to proceed with the installation and configuration of the five international projects. In a second part, low/medium complexity tickets were assigned, where it was already necessary to meet with clients, identify requirements, design the solution, and implement. In the last part of the internship, responsibility was assigned for the maintenance/optimization of an international project, where it was necessary to develop new modules, change existing code (custom evolutionary corrections that emerged from the client), and optimize databases.

Keywords: Information Technologies, Information System, E-Banking, M-Banking, Banking Sector.

Índice Geral

Agradecimentos	iii
Resumo	v
Abstract.....	vii
Índice Geral	ix
Lista de Siglas/Abreviaturas (opcional)	xi
Índice de Figuras	xiii
Capítulo 1 Introdução.....	1
1.1. Objetivo	1
1.2. Enquadramento e relevância	1
1.3. Descrição do estágio	3
1.4. Organização do relatório.....	4
Capítulo 2 Caracterização da empresa	5
2.1. Descrição da empresa	5
2.1. Metodologia de trabalho	6
2.1.1. Metodologia Kanban	7
2.1.2. Criticidade de um Ticket	7
2.1.3. Git Flow.....	8
Capítulo 3 Enquadramento teórico.....	11
3.1. Introdução	11
3.2. Importância das TI no setor bancário	13
3.3. <i>E-Banking</i> e Inovação	14
3.4. Plataformas informáticas como fator diferenciador.....	15
3.5. Mercado das plataformas Informáticas para banca.....	16
Capítulo 4 Ferramentas e tecnologias	17
4.1. Ferramentas e tecnologias utilizadas	17
4.1.1. C#	17
4.1.2. Microsoft Azure.....	17
4.1.3. HTML5.....	18
4.1.4. Visual Studio Professional	18
4.1.1. JavaScript	19
4.1.2. React	19
4.1.3. SQL.....	19
4.1.4. SQL Server Managment Studio.....	20

4.1.5.	IIS Express.....	20
4.1.6.	ASP.Net.....	20
4.1.7.	Jira	20
4.1.8.	Umbraco	21
4.1.9.	Fiddler.....	21
Capítulo 5	Trabalho efetuado	23
5.1.	Introdução e enquadramento.....	23
5.2.	Atividades desenvolvidas	24
5.2.1.	Instalação dos projetos dos clientes.....	24
5.2.2.	Alteração <i>Call Center</i> e <i>Chat Center</i> no website do cliente	28
5.2.3.	Importação de ficheiros CSV de grandes dimensões	29
5.2.4.	Transferências para carteiras digitais	30
5.2.5.	Plataforma de monitorização dos serviços	31
5.2.6.	Alteração de funcionalidade de abertura de conta.....	32
5.2.7.	Melhoria do sistema de monitorização da base de dados.....	34
5.3.	Formação extra	36
Capítulo 6	Conclusão	39
6.1.	Reflexão sobre o contributo do estágio para o desenvolvimento de competências	39
6.2.	Considerações finais	41
Bibliografia.....		43

Lista de Siglas/Abreviaturas (opcional)

AI	Artificial intelligence
APP	Aplicação
ATM	Automated Teller Machine (Máquina Multibanco)
CRUD	Create, Read, Update e Delete
CSS	Cascading Style Sheets
E-banking	Electronic (Online) banking
ESTiG	Escola Superior de Tecnologia e Gestão.
HTML	Hyper Text Markup Language
IDE	Integrated Development Environment
M-banking	Mobile banking
MySQL	My Structured Query Language
TI	Tecnologias de Informação

Índice de Figuras

Figura 1: Número de utilizadores de serviços <i>online-banking</i> (Fonte: statistica.com)	3
Figura 2: Logotipo da ITSector	6
Figura 3: Esquema de trabalho Kanban utilizado pela equipa.	7
Figura 4: Fluxo de Trabalho do GitFlow (Fonte: www.atlassian.com).....	9
Figura 5: Utilização dos serviços <i>online banking</i> / Reino Unido (fonte: statistic.com). 12	
Figura 6: Mercado das aplicações para a banca 2020 (fonte: appsruntheworld.com)....	16
Figura 7 : Componentes Individuais instalados no Visual Studio.....	25
Figura 8 : Visão Geral da ferramenta IIS.	25
Figura 9 : Janela de Clone do Projeto no Visual Studio.....	26
Figura 10 : Application Pools criadas no IIS.....	26
Figura 11 : Exemplos presentes no ficheiro <i>hosts</i>	27
Figura 12 : Estrutura do ficheiro Web.Config após alteração.	27
Figura 13 : Demonstração do resultado esperado da página.	28
Figura 15 : Demonstração da Pop-Up retornada na página inicial do Website.....	29
Figura 16: Excerto do Web.Config do Internet-Banking	30
Figura 17: Campos da Tabela anteriormente á alteração.	31
Figura 18: Demonstração da Plataforma de Monitorização.	32
Figura 19: Resultado esperado pelo cliente.	33
Figura 20: Excerto de código referente á filtragem do serviço.	33
Figura 21: Tabela Genérica criada no BackOffice do cliente	33
Figura 22: Tabelas criadas na base de dados.	34
Figura 23: <i>Procedure</i> referente à limpeza de uma das tabelas de monitorização.	35
Figura 24: <i>Procedure</i> referente à limpeza de uma das tabelas de histórico.	35
Figura 25: Listagem de algumas das <i>Procedures</i> criadas.	36
Figura 26: Listagem de todos os Jobs criados.	36

Capítulo 1 Introdução

1.1. Objetivo

O estágio terá como principal objetivo o suporte e correção evolutiva de plataformas especializadas para o setor bancário. A sua manutenção pode passar por diversas instâncias, como: a) correções de incoerências no código; b) novos desenvolvimentos consoante as novas necessidades que possam surgir por parte do cliente; e c) melhorias de desempenho nas diversas camadas da aplicação. As tarefas a realizar durante o estágio estão dependentes dos pedidos (*tickets*) efetuados pelos clientes das plataformas (entidades bancárias).

1.2. Enquadramento e relevância

Grande parte dos produtos bancários atuais já existiam nos anos 90. Face à pouca diferenciação da oferta entre concorrentes, a grande preocupação que se formou foi a de gerar competitividade através de um melhor serviço ao cliente. Desta forma, o aparecimento das plataformas especializadas para o setor bancário revelou-se superior comparativamente com o método tradicional. O aparecimento destas soluções tornou o procedimento tradicional muito pouco eficiente e em declínio, visto que todos os processos dependem de colaboradores e estar presente fisicamente nos balcões necessários para o efeito. Estas tecnologias podem utilizar dados em nuvem, análise de *big data*, mecanismos de busca e muitas outras tecnologias (Jihene Ouakdi, 2018).

Com a evolução das TI, a transformação digital deste setor proliferou, dando origem ao denominado “*e-banking* ou *online banking*”, que consiste num sistema que permite que as partes interessadas (por exemplo: clientes e empresas financeiras), tenham acesso às suas contas bancárias e operações bem como a informação acerca de produtos e serviços financeiros via eletrónica (Raymond, 2018). O *online-banking* é a ligação entre o cliente e o banco através da Internet, esta ligação pode ser estabelecida através de um computador (*Internet/Home Banking*) ou através de um dispositivo móvel (*Mobile banking*).

Os serviços *home/mobile banking*, disponibilizados pela generalidade da banca permite aproximar o cliente com o banco, possibilitando assim que as instituições beneficiem de redução de custos potenciando as receitas (Sripalawat, 2011). À banca tradicional, os serviços *e-banking* permitiram: a) reduzir os custos operacionais devido à diminuição do número de balcões e de recursos humanos; e b) reduzir o tempo de espera, resultando num aumento de desempenho de vendas e num maior alcance global (Raymond, 2018). Aos clientes, o *e-banking* permite realizar um conjunto de operações bancárias por via eletrónica, a qualquer hora e em qualquer lugar, e desta forma os clientes já não estão limitados ao horário dos balcões, nem sendo necessárias deslocações ou filas de espera.

Desta forma, é crítico para as empresas desenvolverem novas estratégias de chegar aos clientes através deste meio. O papel do *online-banking* é sem dúvida crítico e a constante inovação das plataformas tem permitido mudar radicalmente o modelo de negócio e os hábitos da população. Segundo o INE (INE, 2019), em 2018, 52% dos utilizadores de Internet dos 16 aos 74 anos de idade referem gerir a conta bancária através do acesso online (*internet banking*), mais 10 p.p. que no ano anterior, mas ainda em nível inferior ao que tem vindo a ser registado na UE-28. Em 2021, nos Estados Unidos, 64,6% dos cidadãos utilizavam *online banking* e 80% prefere este tipo de serviço.

Como podemos verificar na figura seguinte, o número de utilizadores destes serviços irá crescer significativamente nos próximos anos.

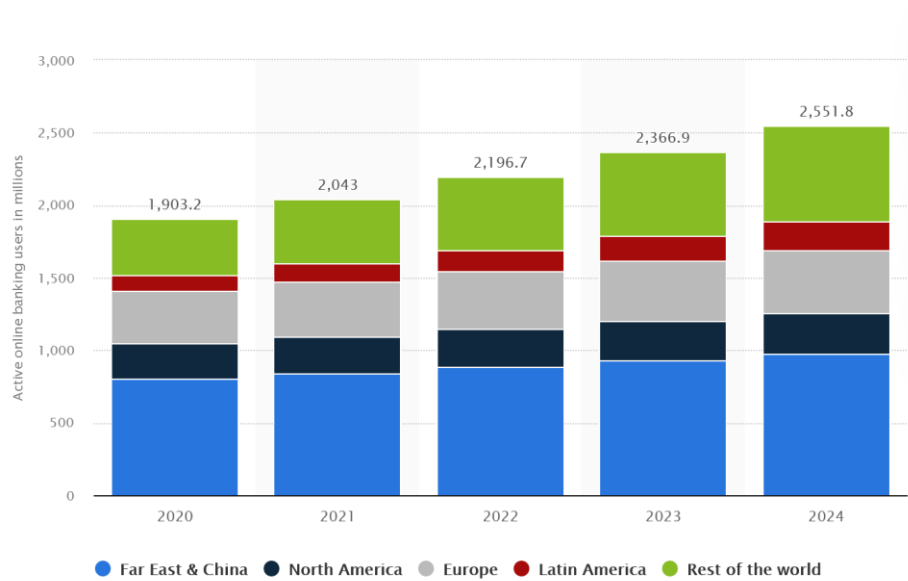


Figura 1: Número de utilizadores de serviços *online-banking* (Fonte: statistica.com)

Neste contexto, a implementação e manutenção de plataformas à medida do cliente torna empresas de TI vitais neste contexto atual.

1.3. Descrição do estágio

O presente relatório descreve o trabalho realizado, ao longo do estágio profissional realizado no âmbito do segundo ano do mestrado de Informática, na empresa ITSector (01/09/2021 a 31/03/2022), na área de suporte a plataformas informáticas do setor bancário.

Na primeira parte do estágio foi necessário proceder à instalação e configuração dos cinco projetos internacionais atribuídos.

Numa segunda parte, foram atribuídos tickets de baixa/media complexidade, onde já foi necessário reunir com os clientes, identificar requisitos, desenhar a solução, e implementar.

Na última parte do estágio, foi atribuída a responsabilidade pela manutenção/otimização de um projeto internacional, onde foi necessário desenvolver novos módulos, alterar código existente (correções evolutivas personalizadas que surgiram por parte do cliente), e otimizar bases de dados.

1.4. Organização do relatório

Quanto à estruturação, o presente relatório irá ser dividido em seis capítulos. De seguida procedemos à sua explicação sucinta:

- No capítulo 1 é realizada a abordagem introdutória às motivações, objetivos, enquadramento e descrição sumária do trabalho realizado no estágio.
- No capítulo 2, é dada a conhecer a empresa onde decorreu o estágio e descrição das metodologias de trabalho aplicadas.
- No capítulo 3, realizar-se-á o enquadramento teórico das temáticas no âmbito do estágio relativamente a Inovação, *E-Banking* e Importância a das TI no setor bancário.
- No capítulo 4, são abordadas as tecnologias e ferramentas utilizadas no âmbito do estágio.
- No capítulo 5 contém o trabalho realizado ao longo das mil cento e sessenta horas de estágio na ITSector.
- No último capítulo são evidenciadas as conclusões finais focando os contributos do estágio para o desenvolvimento de competências.

Capítulo 2 Caracterização da empresa

2.1. Descrição da empresa

A ITSector é uma empresa centrada no desenvolvimento de software especializado para o setor bancário, onde possibilita a utilizadores em todo o mundo acederem facilmente aos seus bancos. A ITSector foi adquirida pelo Grupo Alten em 2020, empregando 37 mil pessoas e está presente em 32 países.

Fundada em 2005, conta com cerca de 600 colaboradores (em Portugal) a trabalhar em conjunto para criar soluções personalizadas para os mais diversos negócios. Foi criada para fornecer ao mercado de TI soluções de alto valor em projetos de transformação digital, crédito e desenvolvimento especializado. Com presença a nível nacional em Porto, Lisboa, Castelo Branco, Braga, Aveiro e Bragança, conta também com expansão internacional em Moçambique, Angola, Alemanha e Polónia.

A ITSector desenvolve soluções de software personalizadas como Plataformas Bancárias, Soluções de Crédito e Aplicações Móveis Avançadas, para iOS e Android. As Soluções de Crédito oferecem total controlo nos respetivos processos. Por parte da ITSector, a visão *Omnichannel* proporciona aos utilizadores aceder a qualquer hora, em qualquer sítio, em qualquer dispositivo e através de qualquer canal às várias soluções. Estas soluções são fundamentais dada a elevada disponibilidade e procura, tornando a performance um aspeto diferenciador.

Os Projetos de Desenvolvimento de Software da ITSECTOR foram classificados no nível 2 de maturidade no modelo CMMI-DEV v1.3, bem como certificados em conformidade com o Sistema de Gestão de Segurança da Informação com os requisitos da ISO 27001.

A ITSector tem um vasto leque de tecnologias e plataformas que utiliza como base, das quais inclui .NET, React, SQL, SharePoint, Mulesoft, Angular, Java, IOS, Android e Windows. A empresa tem mais de 200 projetos em mais de duas dezenas de países, faturando um total de 12,5 milhões em 2016, tendo elevado a fasquia para os 20 milhões até 2020. Por sua vez, o Grupo ALTEN nasceu em 1988 e realiza projetos de investigação e desenvolvimento para empresas líderes em vários setores. Está cotado na bolsa de Paris, e atingiu, em 2019, um volume de negócios de 2,6 mil milhões de euros.

Trata-se de uma empresa inovadora no setor financeiro, dirigindo as suas competências para ajudar bancos e ‘*credit unions*’ a implementar as suas transformações no digital. Mais do que nunca, e com o efeito pandémico, os clientes das instituições financeiras querem ver as suas necessidades satisfeitas sem ter de recorrer ao contacto físico, e beneficiando de experiências simples, seguras e orientadas ao seu perfil (Jornal de negócios /2021).



Figura 2: Logotipo da ITSector

2.1. Metodologia de trabalho

De forma a agilizar resolução de tópicos e possíveis correções evolutivas relacionados com o produto desenvolvido, é necessário promover uma eficiente e cordial metodologia de trabalho e comunicação entre o cliente e a equipa de suporte e que sejam seguidos os pontos importantes para o fluxo do mesmo. O fluxo de resolução de um tópico pode por vezes envolver bastantes camadas de código e serviços, bastando uma ausência mínima de informação para poder originar uma maior dificuldade em encontrar a sua.

2.1.1. Metodologia Kanban

A metodologia primordial utilizada pela equipa, de forma a gerir todo o trabalho, passa pela metodologia *Kanban*. Pode ser definida como um método ágil que se foca em tópicos de correção evolutiva e numa gestão flexível de tarefas, permitindo uma alteração constante dos focos da equipa consoante as prioridades dos temas que possam surgir (Brechner, 2015). Esta metodologia é adaptativa, garantindo que a quantidade de trabalho necessária seja igual às capacidades de trabalho da equipa (Campbell, 2021).

Esta metodologia foi criada em 1940 pela Toyota (Japão) como um sistema de planeamento simples, cujo objetivo era controlar e gerir o trabalho e o stock em todas as etapas da produção de maneira otimizada.

Uma das principais razões para o desenvolvimento do *Kanban* foi a falta de produtividade e eficiência da Toyota em comparação com seus rivais americanos. Com o *Kanban*, a Toyota alcançou um sistema de controlo de produção “*just-in-time*”, flexível e eficiente que aumentou a produtividade ao mesmo tempo que reduzia o custo intensivo do stock de matérias-primas, materiais semiacabados e produtos acabados (Brechner, 2015).

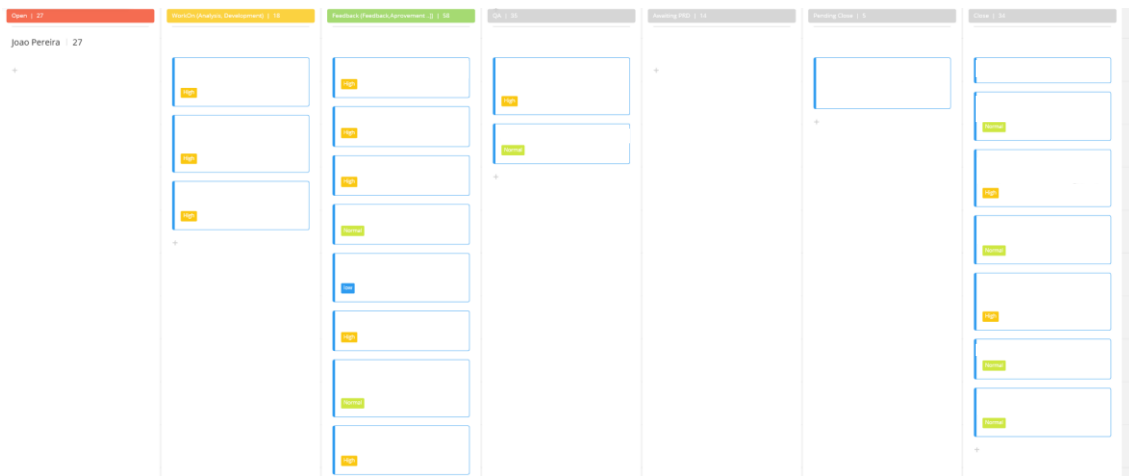


Figura 3: Esquema de trabalho Kanban utilizado pela equipa.

2.1.2. Criticidade de um Ticket

A prioridade definida inicialmente na criação do Ticket define a organização da equipa para a sua resolução. Permite definir a ordem de resolução dos tickets abertos, privilegiando as prioridades mais altas, e a alocação da equipa a cada um deles. Sendo

assim, uma correta prioridade na criação do ticket torna-se num ponto importante, definindo no final de contas quais temas serão mais urgentes e prioritários.

Podemos destacar três níveis de criticidade e respetivas prioridades:

- Criticidade 1 (Prioridade *Emergency*): Quando o incidente inviabiliza o funcionamento por completo de uma aplicação.
- Criticidade 2 (Prioridade *High*): Quando o normal desempenho de uma aplicação é afetado.
- Criticidade 3 (Prioridade Normal): Quando o incidente não inviabiliza a realização de operações na aplicação.

Tendo em conta as descrições de cada um dos níveis, a criação de tickets com estes critérios irá beneficiar todo o fluxo, permitindo uma melhor organização interna do suporte e mais eficazes.

2.1.3. Git Flow

De forma a existir um devido desenvolvimento/manutenção da plataforma é necessário que exista um fluxo de trabalho para o repositório GIT, que permita uma organização continua do que foi realizado.

Tanto uma correção evolutiva, como uma correção no âmbito do contrato de manutenção ou até um desenvolvimento de raiz é necessário que seja disperso por várias camadas de *Branches* dependentes dos seus ambientes e do sucesso dos testes realizados.

A figura seguinte mostra um exemplo de um fluxo de trabalho do Git Flow. Este fluxo irá passar sequencialmente pelas seguintes *Branches* de código:

- Branch Master.
- Branch Develop
- Branch Feature
- Branch Hotfix
- Branch Release

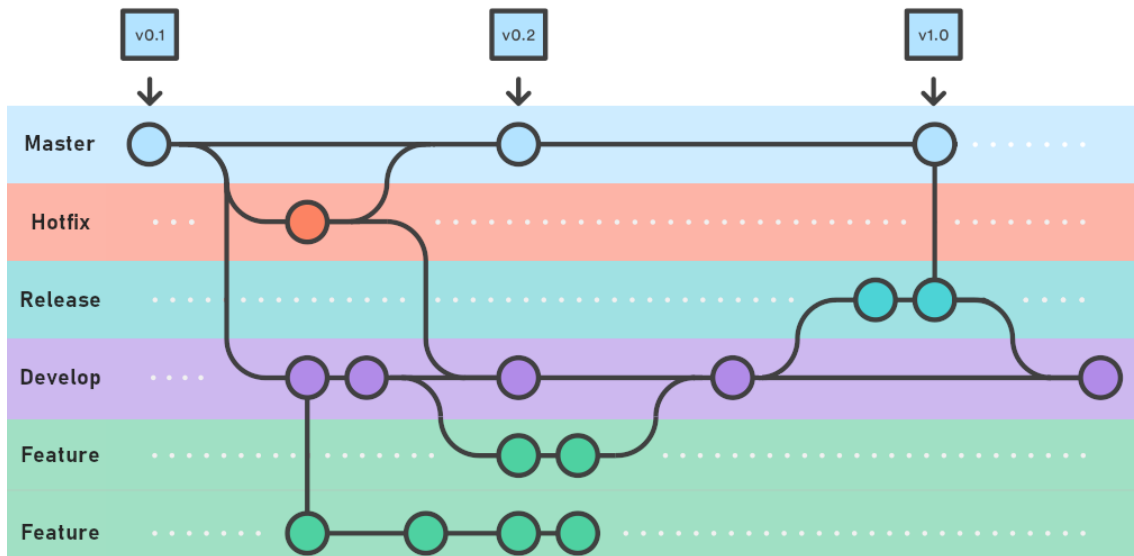


Figura 4: Fluxo de Trabalho do GitFlow (Fonte: www.atlassian.com)

Branch Master

A *Branch Master* contém sempre um reflexo do código que se encontra no ambiente de Produção, além de que irá facilitar o acesso a versões mais antigas neste ambiente. Quando existe um *hotfix* (correção/correção evolutiva) ou uma nova *release*, deveremos criar uma *branch* com base Master para proceder á devidas implementações.

Branch Develop

A *Branch Develop* sedia os últimos desenvolvimentos realizados no projeto. Baseia-se na *Branch Master*, mas com algumas funcionalidades que ainda não foram para o cliente. Podemos então definir que se trata de uma base para o desenvolvimento de novas *features*.

Branch Feature

A *Branch Feature* trata-se de uma *branch* temporária, que funciona com auxílio no fluxo de trabalho, visto que definem novas funcionalidades específicas para a plataforma. Uma nova *feature* começa e termina obrigatoriamente a partir da Branch Develop.

Branch Hotfix

A *Branch Hotfix* refere-se a uma *branch* auxiliar, utilizada quando surgem tópicos que seja necessária intervenção e correção imediata no ambiente de produção.

Branch Release

A *Branch Release* trata-se do lançamento de uma nova versão da plataforma. Nela é incluída todos os pontos concluídos de desenvolvimentos e *hotfixs* pretendidos.

Capítulo 3 Enquadramento teórico

3.1. Introdução

Nos anos 80 e 90, registaram-se grandes alterações e mudanças no setor bancário, principalmente na Europa Ocidental, devido à crescente importância na atividade económica e à alteração na sua forma de atuação (Silva, 2003). As principais mudanças a nível tecnológico surgiram com o aparecimento e expansão dos ATM (*Automatic Teller Machines*), levando a uma diminuição de tarefas rotineiras que até então eram executadas por funcionários. Portugal foi um dos últimos países a adotar as ATMs e as operações baseadas em cartão, mas que permitiu aprender com todas as outras experiências que desenvolveram as suas redes tendo como alvo as ATMs em si, mas criando uma rede de ATMs: Multibanco. A partir de 1985, com 29 apenas 12 máquinas ATM, principalmente em Lisboa e Porto, a SIBS levou a tendência inovação e continuou a evoluir a rede Multibanco (Reis, 2017). Esta evolução tecnológica levou ao início de um novo modelo de negócio e à redução drástica de agências.

A massificação da internet e o menor custo dos computadores permitiu às empresas chegar a um maior número de consumidores, com menores custos. Atualmente, o sucesso dos bancos é regido pela capacidade de adaptação e rapidez de resposta à necessidade de reinventarem o seu modelo de negócio e proposta de valor para os clientes (Motiwalla, 2019). A utilização das TI ao longo dos anos no setor da banca, desde as caixas automáticas ao *Internet banking (home banking)* e *mobile banking/APP* (e meios de pagamento eletrónico), têm mudado completamente o paradigma de negócio, proposta de valor para os clientes e a forma de trabalhar. O uso do *home/mobile banking* por parte dos clientes são as recentes tendências e onde tem existido mais foco.

O serviço *mobile*, disponibilizado pela generalidade, da banca permite aproximar o cliente com o banco, utilizando diversas opções disponíveis, permitindo assim que as instituições beneficiem de redução de custos potenciando as receitas (Sripalawat, 2011). Uma das vantagens de utilizar estes serviços prende-se com o fato de o serviço *mobile* melhorar o atendimento ao cliente, a gestão financeira está acessível em qualquer lugar e a qualquer hora. Este serviço é o que tem a maior percentagem de crescimento.

Em 2017, a utilização da internet para aceder a serviços bancários arrecadou mais adeptos em Portugal, havendo mais de 2,5 milhões de portugueses a utilizarem regularmente a banca online (fonte: Jornal de Negócios). Os dados do 1º trimestre da MasterCard indicam que 21,70% dos portugueses pagam as suas compras através da app do *home banking* e continuam a aumentar. Por exemplo, em 2021, cerca de 67% dos contactos dos clientes do Novo Banco são feitos através dos canais digitais, com grande destaque para o canal *mobile* (53%) (Fonte: jornaldenegocios.pt). O contexto pandémico reduziu as visitas aos balcões (sendo também um dos objetivos dos bancos de forma a reduzir o número de funcionários), enquanto o acesso e a utilização diária dos canais digitais subiram e, em média, cada cliente digital acede pelo menos uma vez por dia à sua app.

No caso do Reino Unido, há uma projeção de que existirá uma taxa de utilização superior a 90% em 2022 (ver figura seguinte).

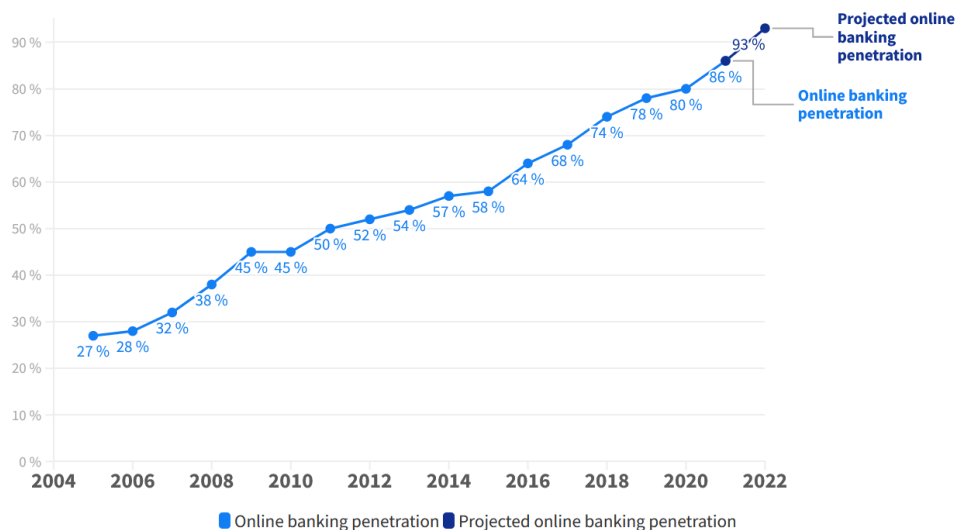


Figura 5: Utilização dos serviços *online banking* / Reino Unido (fonte: statistic.com)

3.2. Importância das TI no setor bancário

O setor bancário tem sido alvo de alterações tecnológica e, de forma gradual, novas formas de distribuição de informação aos consumidores e dos serviços foram introduzidas, permitindo aos consumidores realizar operações financeiras com recurso a plataformas online ou mobile (Raymond, 2018). Atualmente a informatização de todas as transações e processos bancários tornou-se num ponto fulcral no setor, seja no âmbito da eficiência do atendimento ao cliente como na possibilidade de aceder em qualquer lugar. O *e-banking* é um *e-service* prestado pela banca, derivado da transformação e inovação tecnológica. Esse termo pode ser entendido como qualquer atividade bancária realizada com recurso a TI (Motiwalla, 2019)

Com os constantes desenvolvimentos na área das TI, as transações financeiras deixaram de exigir a presença física. Um cliente pode efetuar um conjunto de operações bancárias sem ter de se deslocar a uma agência bancária, levando ao constante desaparecimento de balcões físicos (Reis, 2017) (Ganguly, 2015). Os serviços e ações que habitualmente eram fornecidos através de uma agência bancária (presença física) e que agora são oferecidos a qualquer pessoa através de meios de acesso eletrónicos ou digitais (Schindler, 2019). O custo das operações nos canais digitais é bastante inferior aos custos associados aos balcões, pelo que com uma densidade de penetração de canais físicos como a existente em Portugal, os bancos procuram gradualmente reduzir o número de balcões. Desta forma, a implementação e manutenção de plataformas à medida do cliente torna empresas TI bastante importantes neste contexto atual.

Como referido anteriormente, os serviços de *e-banking* têm vantagens que vão de encontro ao estilo de vida atual dos consumidores: transações mais rápidas, comodidade, facilidade de acesso e uso, quebrando barreiras de espaço e tempo (Jihene Ouakdi, 2018). Por outro lado, o problema destas soluções é a grande dependência tecnológica de conectividade. As principais desvantagens do uso de serviços bancários online encontram-se nas restrições de acesso devido a condicionantes (falhas de ligação, atualização e manutenção de bases de dados e sistemas), proteção de dados, demora de abertura até ativação do serviço e a resolução de problemas (Deloitte, The Deloitte Global Millennial Survey 2019, 2019).

A análise do comportamento dos clientes tem demonstrado a transição do *home banking* para *mobile banking* (Zhou, 2018). Além de todas as características referidas

anteriormente, estes tipos de serviços financeiros disponíveis para smartphones permite que os utilizadores estejam confortáveis com a sua segurança que é garantida com passwords, sejam biométricas ou não.

3.3. E-Banking e Inovação

A banca está entre as indústrias que apostam mais em inovação, sendo a última tendência, em automatização bancária, a mobilidade dos serviços. Esta informatização dos serviços bancários e expansão dos canais remotos, iniciou-se com a introdução dos caixas de pagamento automáticas (ATM) e evoluiu para o *Internet banking*, seguindo-se o *mobile banking*. Neste contexto, os bancos foram-se tornando progressivamente móveis e acessíveis em qualquer lugar por meio da ligação da tecnologia dos telemóveis com a Internet (Schindler, 2019).

A inovação que se deu no setor financeiro conduziu a que o *online banking* influenciasse a utilização tradicional dos serviços bancários, nomeadamente nas deslocações aos bancos, permitindo aos utilizadores poupar tempo, e realizar as operações com maior rapidez e facilidade (Stojanović, 2019). Com o surgimento e desenvolvimento de um mercado online mais extenso, deram-se também alterações do ponto de vista do comportamento dos consumidores. Neste contexto, o *mobile banking* afigura-se como um método de *internet banking* que ganhou uma maior atenção por parte dos consumidores como resultado da facilidade de utilização, tornando-se uma nova fonte de rendimento para os bancos (Motiwalla, 2019)

A inovação no setor bancário possui uma alta probabilidade de falha devido à sua dependência em condições inesperadas. O desenvolvimento bancário pode afetar positivamente a probabilidade de inovação, particularmente para empresas em setores mais dependentes de financiamento externo, e onde os mercados financeiros não são desenvolvidos (Jihene Ouakdi, 2018) Segundo o estudo (Gopane, 2018), o desenvolvimento bancário pode levar à inovação a nível empresarial por meio de dois canais. O primeiro canal é a capacidade de todos os intermediários financeiros beneficiarem da recolha de dados e permitir reduzir os custos de triagem e monitorização, com o conseqüente aumento na eficiência de alocação de recursos, e o segundo canal o crescimento, tornando o *E-banking* como um fator primordial de inclusão.

3.4. Plataformas informáticas como fator diferenciador

A elevada concorrência no setor bancário tem levado os bancos a apostar cada vez mais em plataformas informáticas (*web/mobile*) para se conseguirem diferenciar e atrair clientes. Além da importância de atrair novos clientes, a capacidade de reter clientes é cada vez mais importante. Segundo o estudo Deloitte (Deloitte, The Deloitte Global Millennial Survey 2019, 2019), o custo de captar um novo cliente é dez vezes superior à sua manutenção. A angariação de novos clientes não deixa de ser importante, contudo muitas empresas têm colocado na linha da frente o interesse em manter os seus clientes atuais, intensificando a sua relação (Motiwalla, 2019). Neste sentido, as plataformas atuais disponibilizam, também, aos bancos canais de promoção e divulgação.

No setor da banca existe uma constante necessidade de mostrar aos seus clientes, que são os pioneiros na inovação e os primeiros a fornecer determinado produto. Cada vez mais este canal possibilita uma comunicação entre as duas partes, fornecedor e cliente, podendo este último através da Internet solicitar informações, fazer reclamações ou sugestões (Reis, 2017).

Os dados armazenados sobre os clientes (*bigdata*) permitem fornecer especificamente os produtos e serviços de uma forma mais direcionada, evitando sobrecarregar o cliente com produtos sem interesse para o próprio. As plataformas, disponibilizam ferramentas de *data-mining* e *Artificial intelligence* (AI) que permitem extrair padrões de comportamento e executar tarefas que anteriormente requeriam intervenção humana e processos extremamente demorados - Identificar os clientes e seus hábitos de consumo, ajuda o banco na estratégia de venda de novos produtos e serviços, permitindo assim que as instituições beneficiem de redução de custos potenciando as receitas.

Desta forma, a digitalização da relação com o cliente irá gerar uma enorme quantidade de dados que constituirão a pegada digital dos clientes fazendo do elemento digital e da análise de dados as novas prioridades (Motiwalla, 2019). A utilização de grandes quantidades de dados para a deteção de padrões de consumo (potencial ou efetivo) possibilita a antecipação das necessidades de vários consumidores, trazendo novos desafios e oportunidades, isto é a utilização de *Big Data*. Estas grandes quantidades de dados permitem às instituições adaptar e melhorar o seu leque de produtos, por exemplo, através de ajustamentos à localização geográfica. Assim, temos vindo a assistir a grandes investimentos em sistemas de informação que permitam ter uma visão global dos seus

clientes, através da utilização de *scorings* e monitorização da inclinação para certos produtos financeiros, sendo uma peça-chave nos dias de hoje (Ganguly, 2015). Será através destes dois elementos-chave da estratégia do setor bancário que se irão desenvolver os mecanismos que têm estado na agenda dos gestores em qualquer setor de atividade: redução de custos, conquista de novos clientes e desenvolvimento da relação com os existentes (Stojanović, 2019).

3.5. Mercado das plataformas Informáticas para banca

O mercado de soluções informática para abanca engloba muitas empresas de média e grande dimensão (nacionais e estrangeiras). A nível mundial, empresas como a Microsoft, Oracle, SAP, e FIS global são as que possuem maior cota de mercado (ver figura seguinte).

Segundo a consultora Deloitte (Deloitte, Blockchain for Financial Leaders: Opportunity vs. Reality, 2020), em 2020 o top 10 dos vendedores de software para o setor bancário detinha aproximadamente 50.9% do mercado o que correspondia a 32,6 mil milhões de dólares.

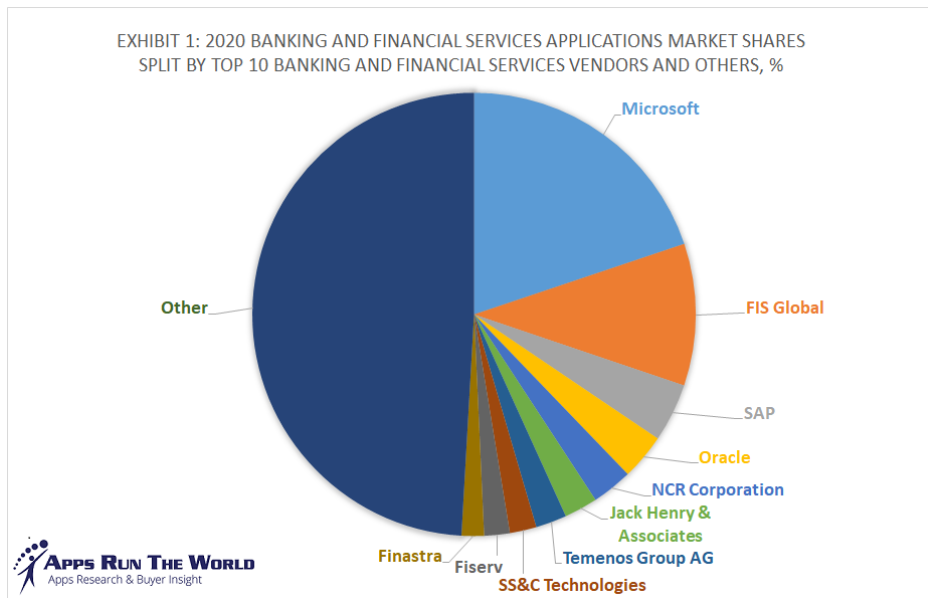


Figura 6: Mercado das aplicações para a banca 2020 (fonte: appsruntheworld.com)

Capítulo 4 Ferramentas e tecnologias

4.1. Ferramentas e tecnologias utilizadas

Durante o estágio foram utilizadas várias ferramentas para o desenvolvimento e manutenção da aplicação e base de dados. As ferramentas empregues nos projetos foram apresentadas pela empresa onde se realizou o estágio como referências na área.

4.1.1. C#

Trata-se de uma linguagem de programação orientada a objetos e permite que se crie componentes reutilizáveis para uma ampla variedade de tipos de aplicações, tendo sido introduzida pela Microsoft em 26 de junho de 2000.

C# é uma evolução da família de linguagens de programação como o C e C++. No entanto, ele empresta recursos de outras linguagens de programação, como Delphi e Java. O conceito da programação orientada a objetos é um dos princípios essenciais das *frameworks* disponibilizadas pelo .Net. Alguns dos conceitos indispensáveis que esta linguagem disponibiliza para todas as tecnologias que usufruam da mesma são a herança, encapsulamento e o polimorfismo. (Nagel, 2021)

4.1.2. Microsoft Azure

O Microsoft Azure é uma plataforma destinada à execução de diversos serviços computacionais remotamente, sendo que toda a sua computação em nuvem é caracterizada como combinação de *Software as a Service* (mais conhecido como modelo SaaS, este modelo é uma forma de distribuição e comercialização de software) com Computação em GRID (modelo de computacional constituído por diversas máquinas, que

podem ser acedidas remota ou localmente, com uma grande capacidade de processamento). Esta plataforma disponibiliza serviços como *software as a service* (SaaS), *infrastructure as a service* (IaaS), *platform as a service* (PaaS) e *functions as a service* (FaaS).

Através deste poder de processamento remoto, esta plataforma disponibiliza diversos serviços baseados em .NET, em que nosso caso foi bastante útil nos testes o SQL Azure Database (possibilitou disponibilizar a base de dados num servidor remoto), e serviços API, que permitiu colocar em nuvem a aplicação e fazer o a gestão da mesma (Microsoft, 2020).

4.1.3. HTML5

HTML5 é a última evolução no padrão que define o HTML. Esta é uma nova versão da linguagem HTML, com novos elementos, atributos e comportamentos, e um conjunto maior de tecnologias que permite a criação de sites e aplicações da Web mais diversos e poderosos.

A semelhança do HTML, o HTML5 trata-se de uma linguagem de marcação que é interpretada nas plataformas web. Permite definir elementos para exibir várias estruturas como títulos, tabelas, listas e elementos de entrada, texto e caixas de combinação. Sendo esta somente uma tecnologia associada a morfologia das páginas web, por vezes vemos complementada a tecnologias como CSS e JavaScript. (Frain, 2020)

4.1.4. Visual Studio Professional

Microsoft Visual Studio é um ambiente de desenvolvimento integrado (IDE) da Microsoft para desenvolvimento de software especialmente dedicado ao .NET Framework e às linguagens Visual Basic (VB), C, C++, C# (C Sharp) e F# (F Sharp). Também é um produto de desenvolvimento na área web, usando a plataforma do ASP.NET, como websites, plataformas web, serviços web e aplicativos móveis. A licença Professional, ao qual recorreremos, permite usufruir de mais recursos do que a edição *Community*, como Code-Lens e Team Foundation Server para gestão de código e colaboração em equipa. Inclui vários serviços do Microsoft Azure para desenvolvimento e teste. (Nagel, 2021)

4.1.1. JavaScript

O JavaScript é uma linguagem de programação que é utilizada para o desenvolvimento de plataformas web, de jogos, entre outras. Esta processa as páginas da web de forma interativa e dinâmica permitindo a implementação de logica em páginas web que não poderiam ser feitos somente a nível de HTML.

JavaScript é uma *interpreted language*, não possui a necessidade de ser compilado o código. Esta linguagem pode ser embutida no código HTML, ou através de arquivos JavaScript que são chamados dentro de um elemento Script. Permite aceder ao *document object model* (DOM) a partir da página HTML, o que possibilita alterar dinamicamente elementos na página do utilizador. (Nagel, 2021)

4.1.2. React

React.js é uma biblioteca JavaScript de código aberto usada para criar plataformas, especificamente para aplicações com o mesmo layout. O React.js pode ser utilizado para criar aplicações mobile (React Native), o que significa ser suportada a reutilização extensiva de código. Então, em simultâneo, podem ser feitas aplicações IOS, Android e Web. O React.js usa binding que é conhecido com ligação de dados unidirecional e uma arquitetura de aplicação chamada Flux, que controla o fluxo de dados para os componentes por um ponto de controlo onde é mais fácil depurar componentes independentes de grandes aplicações React.js. As aplicações React.js são muito fáceis de se testar.

4.1.3. SQL

Structured Query Language (SQL) trata-se de uma linguagem de programação que maioritariamente é usada em bases de dados relacionais ou sistemas de gestão de dados, tendo sido desenvolvida pela IBM no início da década de 1970.

Esta linguagem de programação permaneceu uma escolha consistentemente para vários utilizadores de bases de dados ao longo dos anos, principalmente devido à sua facilidade de uso e à sua eficácia.

4.1.4. SQL Server Management Studio

O SQL Server Management Studio (SSMS) é um software integrado para gerir qualquer infraestrutura SQL, do SQL Server a Base de Dados SQL do Azure. O SSMS fornece ferramentas para configurar, monitorar e administrar instâncias do SQL Server e bancos de dados. Também ser utilizado para criar, monitorizar e atualizar as camadas de dados usados pelas aplicações e criar consultas, jobs e scripts.

4.1.5. IIS Express

O IS Express é uma versão otimizada para desenvolvedores, que facilita o uso da versão mais recente do mesmo para desenvolver e testar sites localmente. Possui diversos recursos como o IIS 7 e superior, além de recursos adicionais projetados para facilitar o desenvolvimento de sites, incluindo:

- Compatibilidade com ASP.NET e PHP.
- Possibilidade de vários utilizadores do IIS Express poderem trabalhar no mesmo computador.

4.1.6. ASP.Net

O ASP.NET trata-se de uma extensão da plataforma .NET com ferramentas e bibliotecas específicas para o desenvolvimento de plataformas Web, fornecendo um modelo de programação e diversos serviços bastante importantes para implementação de aplicações web robustas.

Esta extensão usufrui de protocolos HTTP que possibilita uma comunicação e cooperação bilateral navegador-servidor. (Nagel, 2021)

4.1.7. Jira

Podemos definir o Jira como um software de gestão de tópicos e de projetos, tendo sido desenvolvida pela empresa Atlassian, sendo amplamente utilizada por equipas de desenvolvimento ágil e suporte para analisar bugs, desenvolvimentos e outras tarefas que possam existir.

4.1.8. Umbraco

O Umbraco é um CMS open-source (content management system) construído sobre a framework .NET da Microsoft usando ASP.NET e escrito em C#. Permite uma alta flexibilidade, segurança e escalabilidade na forma como é ilustrado todo o conteúdo na página web (Umbraco, 2022). Disponibiliza aos administradores de conteúdo uma plataforma fácil de usar e uma experiência intuitiva.

4.1.9. Fiddler

É uma ferramenta muito popular de rastreamento do tráfego web. É bastante poderosa, mas extremamente fácil de utilizar. Fiddler é uma ferramenta indispensável que permite aos programadores Web verificarem como um *site* funciona, solucionar problemas e fazer avaliações de desempenho (Lawrence, 2015). É facilmente expansível através da utilização do FiddlerScript ou da criação de extensões .NET. As alterações incluem adições à interface do utilizador do Fiddler, além de ajustar automaticamente o tráfego antes que ele seja enviado ao servidor da Web ou antes que a resposta seja enviada ao cliente. Os recursos também podem ser incorporados nas aplicações .NET via FiddlerCore.

Capítulo 5 Trabalho efetuado

5.1. Introdução e enquadramento

Neste capítulo serão detalhadas todas as atividades realizadas ao longo do estágio de mil cento e sessenta horas na empresa ITSector. No período de 01/09/2021 a 31/03/2022, foram desenvolvidas inúmeras tarefas na área de suporte a plataformas informáticas do setor bancário. Na primeira parte do estágio foi necessário proceder à instalação e configuração dos cinco projetos internacionais atribuídos. Numa segunda parte, foram atribuídos tickets de baixa/media complexidade, onde já foi necessário reunir com os clientes, identificar requisitos, desenhar a solução, e implementar. Na última parte do estágio, foi atribuída a responsabilidade pela manutenção/otimização de um projeto internacional, onde foi necessário desenvolver novos módulos, alterar código existente (correções evolutivas personalizadas que surgiram por parte do cliente), e otimizar bases de dados. Todas as reuniões com os clientes são realizadas em inglês e são monitorizadas pelos superiores hierárquicos.

O estágio curricular centra-se na área de suporte, tendo como atividades e responsabilidades inerentes:

- Resolução de tickets;
- Reunião com clientes;
- Desenho de soluções;
- Implementação de novos módulos;
- Alteração de código existente;
- Criação de documentação.

Praticamente todos os dias surgiam novas tarefas (tickets) que implicaram reuniões com clientes, análise de requisitos, desenho de soluções, implementação e validação. Estas

tarefas vão desde criação de novo código, alteração/atualização de código existente, otimização e alteração de bases de dados, etc. Desta forma, como foram realizadas inúmeras tarefas durante o estágio (a grande maioria confidenciais), optamos por nos focar em sete destas.

5.2. Atividades desenvolvidas

Como referido anteriormente, as atividades desenvolvidas são confidenciais e desta forma só é possível apresentar, de forma resumida, algumas das tarefas realizadas ao longo do estágio.

5.2.1. Instalação dos projetos dos clientes

Software/Extensões

Primeiramente, procedeu-se à instalação, compilação e execução de todas as tecnologias necessárias para o correto funcionamento dos projetos. Pretendeu-se um IDE capaz de executar localmente a aplicação na máquina, possuir acesso ao repositório de código, fosse fácil de trabalhar no projeto e o Visual Studio 2019 Professional foi o escolhido, tendo todos estes fatores.

Em complemento ao Visual Studio 2019 é necessária a instalação de diversas extensões relativas as versões das tecnologias .NET, .NET Framework e ASP.Net existentes nos diversos projetos.

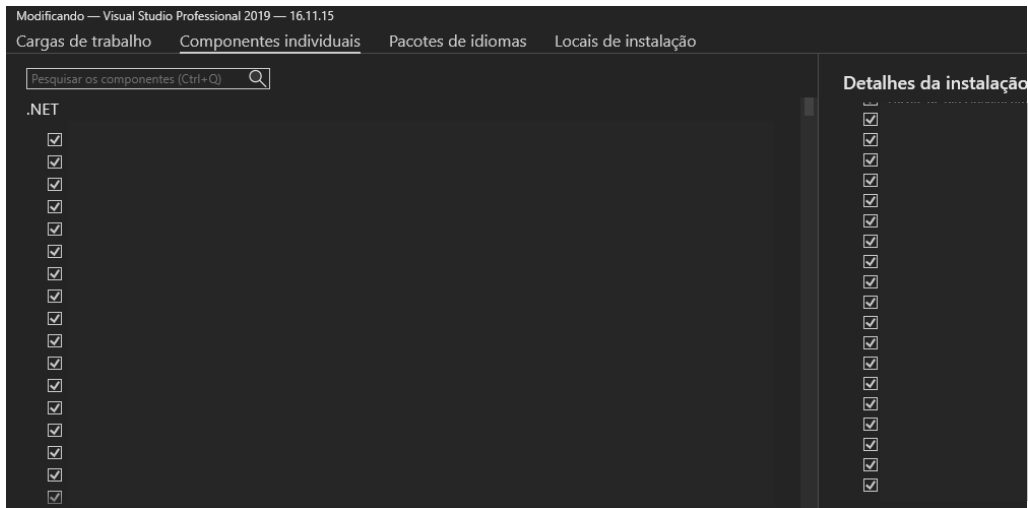


Figura 7 : Componentes Individuais instalados no Visual Studio.

Finalmente foi necessário proceder à instalação do *Internet Information Services (IIS)*, que vai ser o principal interveniente nos serviços locais que a aplicação ira usufruir nos testes.

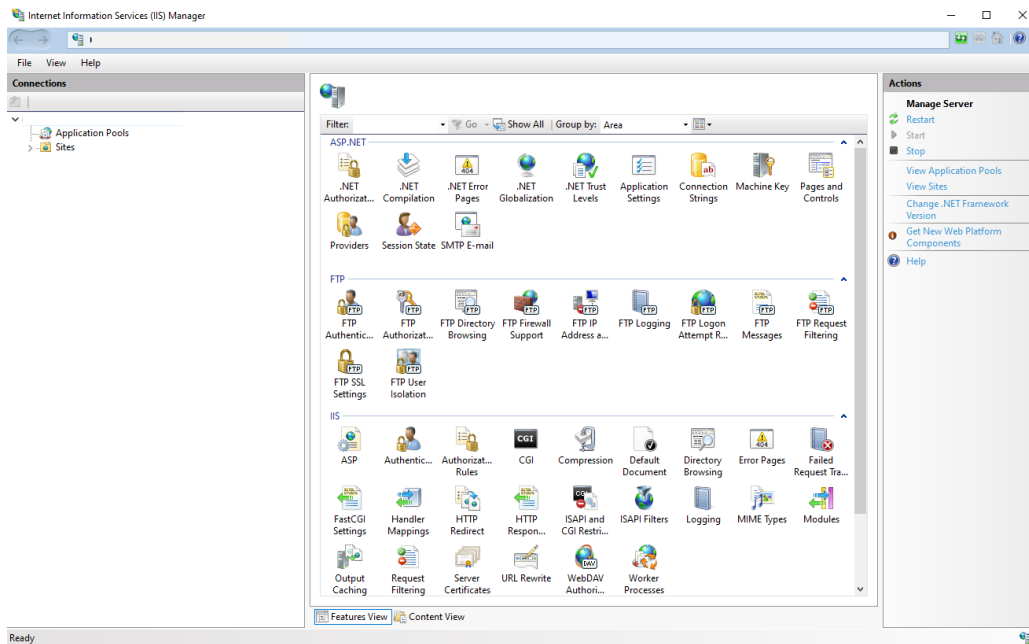


Figura 8 : Visão Geral da ferramenta IIS.

Projeto

Com as tecnologias já devidamente instaladas podemos passar para a colocação do projeto na máquina local. Já no IDE, procedeu-se ao *clone* do projeto diretamente do repositório do Azure DevOps para uma diretoria previamente especificada.

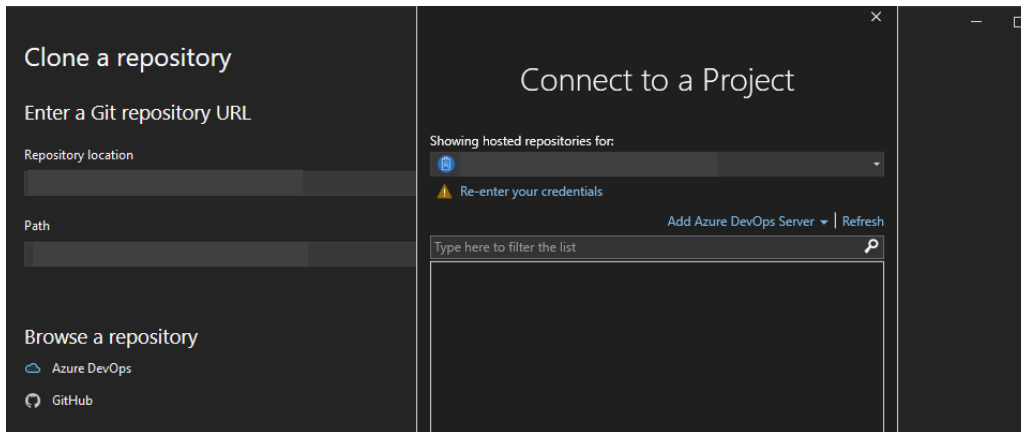


Figura 9 : Janela de Clone do Projeto no Visual Studio.

Configuração

Para compilar e executar o projeto corretamente, é essencial verificar as *Windows Features*, especialmente por causa do WCF e da autenticação de sites.

Já no IIS, procedeu-se a criação dos sites, onde foram preenchidos com as diretorias para o projeto anterior, com os *hostnames* e portas definidas inicialmente na criação do projeto. Estes sites funcionam como um *container* para aplicações e diretorias virtuais, onde permite que sejam acedidos por meio de uma ou mais ligações exclusivas.

Posteriormente em sintonia com os sites criados foi necessário criar as diversas *Application Pools* para cada site. Estas *pools* definem um grupo de um ou mais processos, com configurações idênticas que respondem a solicitações para uma ou mais aplicações atribuídas.

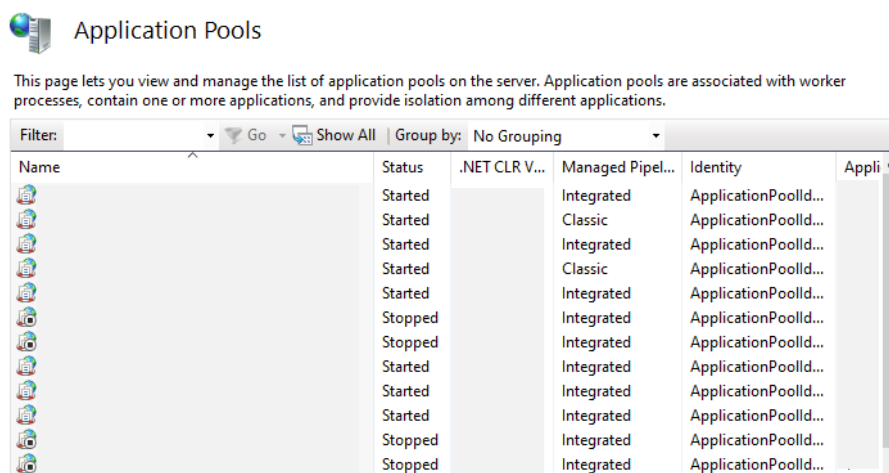


Figura 10 : Application Pools criadas no IIS.

De seguida, foi necessário adicionar localmente no ficheiro *hosts* as regras para os Sites e as *Pools* criadas no IIS. Este ficheiro contém os mapeamentos para endereços IP dos *hostnames*, onde cada entrada deve ser mantida individual. O endereço IP configurado em passos anteriores deve ser colocado na primeira coluna seguido pelo respetivo *hostname*.

```
#
# For example:
#
#      102.54.94.97      rhino.acme.com      # source server
#      38.25.63.10     x.acme.com        # x client host

# localhost name resolution is handled within DNS itself.
#   127.0.0.1          localhost
```

Figura 11 : Exemplos presentes no ficheiro *hosts*.

Finalmente, foi necessário adicionar alguns utilizadores locais com os nomes e password estipulados na criação do projeto com a configuração que a password não ira expirar.

Configurações úteis

O Fiddler, apesar de não se tratar de um passo obrigatório nas configurações do projeto, é uma ferramenta bastante útil para analisar/registar todos os pedidos HTTP efetuados pela plataforma. Procedeu-se a instalação do Fiddler e a configuração da máquina local para capturar todos estes pedidos.

Apos a instalação foi alterado arquivo *web.config*, de forma a contemplar as seguintes definições:

```
<system.net>
  <defaultProxy>
    <proxy bypassonlocal="false" usesystemdefault="true" proxyaddress="http://127.0.0.1:8888" />
  </defaultProxy>
</system.net>
```

Figura 12 : Estrutura do ficheiro *Web.Config* após alteração.

5.2.2. Alteração *Call Center* e *Chat Center* no website do cliente

Descrição da tarefa

No âmbito da alteração da funcionalidade do *Call Center* e *Chat Center* no *website* do cliente, foi desenvolvido um plugin, através da inclusão de *scripts* na página do *website* e recorrendo a um serviço externo, que permite intervenção dinâmica na própria página.

Demonstração do trabalho

De forma manter a estrutura e o correto fluxo da aplicação a proposta consiste em manter os recursos já fornecidos pela plataforma “Umbraco” e pelo próprio fluxo funcional implementado anteriormente na aplicação, sendo necessário proceder também à alteração dos links dos botões no site de recursos do mesmo.

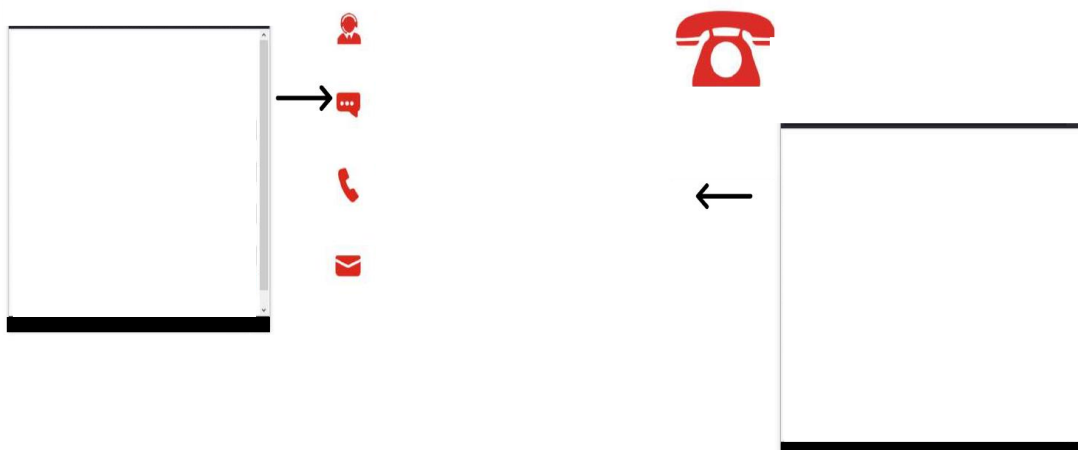


Figura 13 : Demonstração do resultado esperado da página.

Alterando os links e implementando a partir do código já desenvolvido, em conjunto com o Umbraco surgirá ao utilizador uma janela *Popup* à semelhança do já existia previamente, mas com o conteúdo pretendido com as devidas distinções entre o “*Click to Call*” e o “*Click to Chat*”.

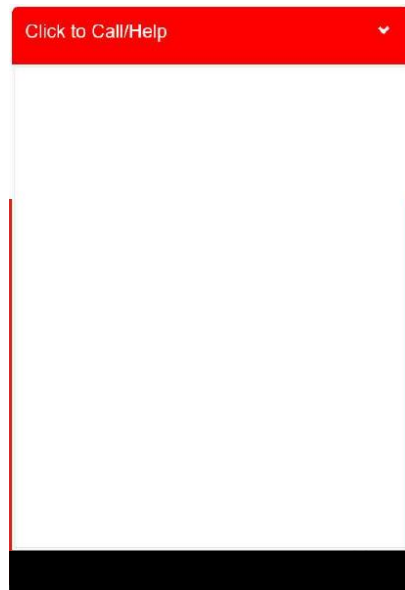


Figura 14 : Demonstração da Pop-Up retornada na página inicial do Website.

Esta funcionalidade encontra-se neste momento em funcionamento em alguns clientes internacionais.

5.2.3. Importação de ficheiros CSV de grandes dimensões

Descrição da tarefa

Devido a um Time-out definido no Web.Config da plataforma *Internet Banking*, ao fim de dois minutos era retornado erro de carregamento de ficheiros csv. Como é normal, o desempenho do serviço regride quanto maior for a quantidade de linhas existente no ficheiro afetando o processamento do mesmo.

Esta alteração no processo de pagamentos via ficheiro focou-se em permitir pagamentos em massa com ficheiros CSV de 5.000 linhas.

Demonstração do trabalho

Nesta melhoria do pagamento em massa tivemos como objetivo: Alterar o processo de carga dos ficheiros csv permitindo carregar ficheiros de até 5.000 em 2 minutos.

```

<binding name="BasicHttpBinding_ICoreService" maxBufferSize="999916384" maxReceivedMessageSize="999916384"
openTimeout="00:02:00" receiveTimeout="00:10:00" sendTimeout="00:02:00" allowCookies="false"
hostNameComparisonMode="StrongWildcard" maxBufferPoolSize="9999524288" messageEncoding="Text" textEncoding="utf-8"
useDefaultWebProxy="true">
  <readerQuotas maxDepth="32" :
  maxNameTableCharCount="999916384" />
  <security mode="None">
    <transport clientCredentialType="None" proxyCredentialType="None" realm="" />
    <message clientCredentialType="UserName" algorithmSuite="Default" />
  </security>
</binding>

```

Figura 15: Excerto do Web.Config do Internet-Banking

Tendo em conta os pontos evidenciados anteriormente, aumentar o *timeout* definido anteriormente para cinco minutos, no entanto, foi feita esta alteração apenas para o *upload*, e não em todos os *Time-Out* definidos.

5.2.4. Transferências para carteiras digitais

Descrição da tarefa

O objetivo desta tarefa foi criar um processo que visa inibir a transferência para carteiras digitais sem impacto no histórico. Antes desta alteração só era permitida a remoção completa da carteira, o que tinha um grande impacto na visualização de dados históricos.

Pretendeu-se criar forma de inibir a transferência para carteiras digitais específicas sem impacto no histórico. A estratégia assentou em:

- Adicionar campo `isVisible` na tabela `Mobile TransferEntity`:
- Alterar serviço que retorna as entidades para não exibir as carteiras digitais que estejam marcadas para não estarem visíveis.

Demonstração do trabalho

Para a realização desta tarefa foi necessário realizar operações a nível da base de dados e utilizar linguagem procedimental. Foi necessário também otimizar algumas consultas de forma a tornar o processo mais rápido para o cliente.

A figura seguinte mostra uma pequena parte de uma tabela utilizada para realizar esta tarefa.

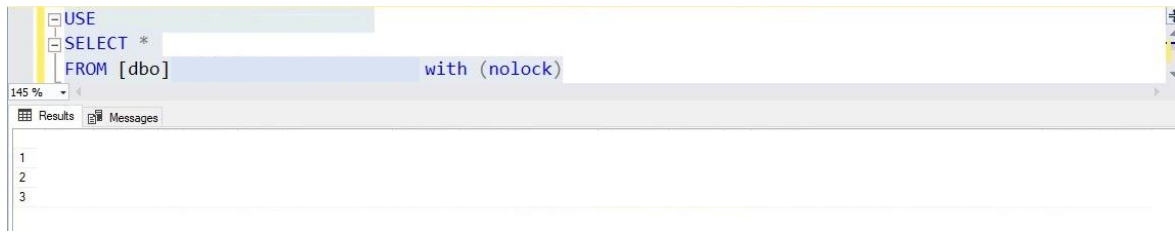


Figura 16: Campos da Tabela anteriormente á alteração.

5.2.5. Plataforma de monitorização dos serviços

Descrição da tarefa

O Banco não possuía nenhuma plataforma de monitorização dos serviços (por exemplo: quando um serviço se encontra lento ou com problemas) o que fazia com que só se tivesse conhecimento dos problemas quando eram reportados.

Desta forma, o cliente pediu uma ferramenta de monitorização capaz de lhe indicar a atual situação dos todos os serviços usufruídos pela plataforma, indicando se os serviços estão disponíveis ou não e a informações importantes, para assim possibilitar uma atuação mais rápida, pontual e eficaz sobre o problema.

Demonstração do trabalho

Tendo em conta o pretendido foi proposta uma solução baseada nos *logs* registados na base de dados (registo de erros ocorridos na plataforma), que serão apresentados num esquema representativo do estado dos serviços em tempo real.

Para refletir o estado dos serviços foi necessário efetuar configurações de testes e em que fosse possível refletir o funcionamento dos mesmos.

Foram associados significados às respetivas cores, permitindo uma rápida identificação do estado atual de cada serviço:

- Verde: Serviço encontra-se a funcionar corretamente;
- Amarelo: Serviço encontra-se bastante lento;
- Vermelho: Serviço encontra-se com erro;
- Cinzento: Serviço nunca foi executado.

A aplicação fornecerá ainda informações uteis relativamente ao:

- Tempo de execução do serviço;
- Número de tentativas de sucesso/falhadas;
- Relatório Estado Serviço (Logs);
- Envio email com relatório de falha.

A figura seguinte mostra um exemplo da plataforma de monitorização implementada, permitindo ao cliente ter uma fácil perceção do estado dos vários serviços.



Figura 17: Demonstração da Plataforma de Monitorização.

5.2.6. Alteração de funcionalidade de abertura de conta

Descrição da tarefa

Esta tarefa surgiu de um pedido do cliente para criar uma nova funcionalidade na abertura de conta de forma a que os utilizadores possam optar por 3 moedas (US Dollar, Euro e GBP). Este pedido deveria ser aplicado apenas na criação do tipo "Conta Poupança".

Demonstração do trabalho

Para responder ao pedido efetuado foi necessário aplicar um filtro no resultado do serviço do cliente. Para isso, foi criada uma tabela com as moedas solicitadas e o respetivo banco poderia alterar, adicionar ou remover as moedas listadas diretamente no *BackOffice* da plataforma. Esse recurso afeta o processo de criação de contas de poupança no *mobile Banking* (IOS e android) e *Internet Banking*.

Na lista de moedas possíveis no formulário Conta Poupança, apenas 3 opções são mostradas conforme solicitado.



Figura 18: Resultado esperado pelo cliente.

Alem da alteração da base de dados, foi desenvolvido um novo modulo que permite executar esta funcionalidade de forma segura e rápida (figura seguinte).

```
var output = CustomBankingP;
input, out res);

if (res != null && res.Currencies.Any())
{
    #region TICKET 9737
    if (this.SelectedProduct.Category == 201) {
        var genericCurrencies = CoreProviderManager.Provider.GetGenericItemsByTableName(CurrentContext.ServiceInputHeader, "CurrencySavingAccountGeneric", VariationHelper.GetCurrentVariationLabelUI(), false);
        List<
            > allowedListCurrencies = new List<
                >();

        if (genericCurrencies != null && genericCurrencies.Any())
        {
            foreach (var item in genericCurrencies)
            {
                var itemcurrency = res.Currencies.Single(r => r.Currency.Equals(item.Value));
                if (itemcurrency != null)
                {
                    allowedListCurrencies.Add(itemcurrency);
                }
            }
            this.ddLCurrencies.DropDown.DataSource = allowedListCurrencies;
        }
        else
        {
            this.ddLCurrencies.DropDown.DataSource = res.Currencies;
        }
    }
}
#endregion
```

Figura 19: Excerto de código referente á filtragem do serviço.

O BackOffice do cliente também foi alterado para uma interface intuitiva de forma a tornar o processo mais eficiente.

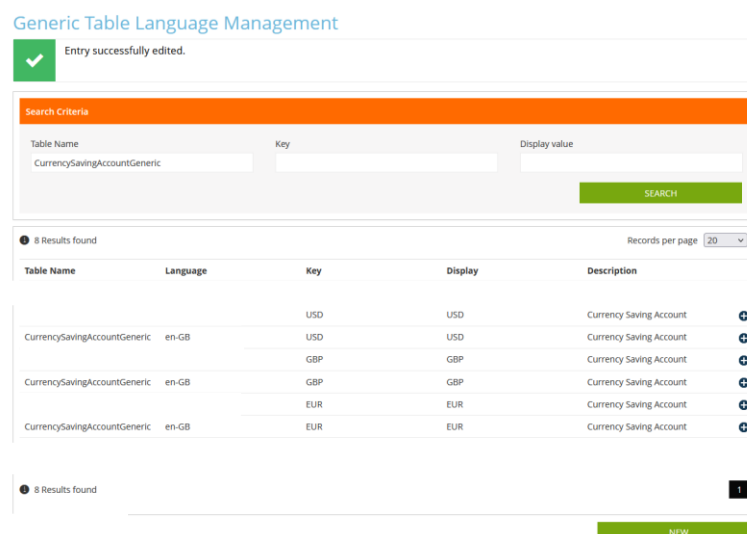


Figura 20: Tabela Genérica criada no BackOffice do cliente

5.2.7. Melhoria do sistema de monitorização da base de dados

Descrição da tarefa

O principal objetivo desta tarefa foi melhorar o sistema de monitorização a nível de base de dados através de um mecanismo de histórico das tabelas já existentes. Este pedido deve ser aplicado nos ambientes de Qualidade e Produção do Cliente.

A melhoria deverá ter em conta os seguintes pontos:

- Criação de tabelas de histórico (espelho das tabelas já existentes de monitorização)
- Criação de várias *procedures (stored procedure)* e *triggers* (linguagem procedimental) que no final do dia removam das tabelas de monitoramento e coloquem nas respetivas tabelas de histórico.
- Criação de *procedures (stored procedure)* e *triggers* (linguagem procedimental) que limpem os dados com mais de 3 meses das tabelas de histórico.
- As limpezas das tabelas de monitorização devem manter somente os dados do dia anterior;
- Criação de um job capaz de executar as *procedures* diariamente à meia-noite

Demonstração do trabalho

A instalação passou em primeira instância por criar as tabelas de Histórico que irão alocar os dados das tabelas principais.

```

+ dbo.Application
+ dbo.ApplicationHeartbeat
+ dbo.Channel
+ dbo.ChannelApplication
+ dbo.DatabaseSizes
+ dbo.Interaction
+ dbo.JsonRequest
+ dbo.JsonRequestHist
+ dbo.LogRequest
+ dbo.LogRequestHist
+ dbo.ServiceRequest
+ dbo.ServiceRequestHist
+ dbo.TableSizes
+ dbo.TimeTable
+ dbo.TransactionExecution
+ dbo.TransactionExecutionHist
+ dbo.WebPageRequest
+ dbo.WebSession
  
```

Figura 21: Tabelas criadas na base de dados.

De seguida, foram executados scripts onde foram criadas as *procedures* que terão como função apagar os dados posteriores ao dia anterior á sua execução e transferir para a sua respetiva tabela de histórico.

```

SET @cycle = 50;

WHILE 1=1
BEGIN

    -- Delete from JsonRequest X rows, and insert them into JsonRequestHist
    -- Atomic operation - No Transaction needed
    WITH TOP_X_ROWS AS (
    SELECT TOP(@cycle) *
    FROM [JsonRequest]
    WHERE datediff(d,[BeginDate],getdate()) > 1)

    DELETE TOP_X_ROWS
    OUTPUT DELETED.*
    INTO [JsonRequestHist]

    -- STOP CONDITION
    IF (@@ROWCOUNT = 0)
        BREAK;
END

```

Figura 22: *Procedure* referente à limpeza de uma das tabelas de monitorização.

Já no âmbito das tabelas de histórico, tiveram de ser geradas as *procedures* que irão retirar do histórico todos os dados com mais de 6 meses, evitando atingir o limite de armazenamento do histórico e consequentemente da base de dados.

```

WHILE 1=1
BEGIN

    -- Delete Top @cycle rows from JsonRequestHist
    -- Atomic operation - No Transaction needed
    WITH TOP_X_ROWS AS (
    SELECT TOP(@cycle) *
    FROM [JsonRequestHist]
    WHERE datediff(d,[BeginDate],getdate()) > @purgeDays)

    DELETE TOP_X_ROWS

    -- STOP CONDITION
    IF (@@ROWCOUNT = 0)
        BREAK;
END

```

Figura 23: *Procedure* referente à limpeza de uma das tabelas de histórico.

Como referido anteriormente, para realizar esta tarefa foi necessário criar vários procedimentos (*procedures*) de forma a otimizar os processos. O recurso ao T-SQL (Transact-SQL: desenvolvida pela Microsoft, é uma *procedural extension* utilizada pelo SQL Server) permite otimizar os processos de forma eficiente.

```

+ [ ] dbo.usp_Purge_JsonRequestHist
+ [ ] dbo.usp_Purge_LogRequestHist
+ [ ] dbo.usp_Purge_ServiceRequestHist
+ [ ] dbo.usp_Purge_TransactionExecutionHist
+ [ ] dbo.usp_SaveJsonRequest2History
+ [ ] dbo.usp_SaveLogRequest2History
+ [ ] dbo.usp_SaveServiceRequest2History
+ [ ] dbo.usp_SaveTransactionExecution2History

```

Figura 24: Listagem de algumas das *Procedures* criadas.

Nesta fase, já com as tabelas de histórico e *procedures* criadas, foi necessário criar quatro *jobs* onde foi colocado a correr diariamente o implementado anteriormente (figura seguinte).

Name	Enabled	Status	Last Run	Next Run	Category	Runnable	Schedu...	Categor...
	yes	Idle	Succe... 19/04/2022 00:00:00	20/04/2022 00:00:00	Database Maintenance	yes	yes	3
	yes	Idle	Succe... 19/04/2022 00:00:00	20/04/2022 00:00:00	Database Maintenance	yes	yes	3
	yes	Idle	Succe... 19/04/2022 00:00:00	20/04/2022 00:00:00	Database Maintenance	yes	yes	3
	yes	Idle	Succe... 19/04/2022 00:00:00	20/04/2022 00:00:00	Database Maintenance	yes	yes	3

Figura 25: Listagem de todos os Jobs criados.

5.3. Formação extra

Durante a realização do estágio curricular foi possível frequentar de diversas formações internas de curta duração no âmbito do “Kanban-Como gerir fluxo contínuo”, “Integração React em Multiplataforma”, “Implementação de aplicações distribuídas utilizando dapr” e “Digital Experience: como desenhar soluções de sucesso”. Com todas estas formações internas foi possível adquirir competências extra que se revelaram importantes a nível profissional.

Foi realizado uma formação em ASP.NET core MVC (.NET 5) onde foi possível aprimorar a criação de plataformas usando esta tecnologia como base.

A nível de base de base de dados, foi utilizado SQL e o Entity Framework para toda a interações com os dados. Abordaram-se tópicos relacionados com a manipulação das

estruturas, estruturas relacionais presentes na base de dados através de migrações do Entity Framework, como adicionar, obter, atualizar e remover informação da base de dados. Envolveu também tópicos relacionados com a criação de serviços para trabalhar com os dados, mas também criar serviços/repositórios genéricos para permitir a implementação de todas as funções CRUD da base de dados.

Finalmente, foi possível abordar pontos importantes relacionados com a renderização de conteúdo dinâmico com ViewComponents, renderização de UI baseada em integração de pagamentos.

Capítulo 6 Conclusão

A realização do estágio permitiu alicerçar os conceitos adquiridos ao longo do mestrado e aprofundá-los em contexto laboral. O objetivo primordial da elaboração do relatório centrou-se na descrição de algumas das atividades desenvolvidas durante o período de permanência na entidade de acolhimento do estágio, no âmbito do mestrado de informática. Além disso, foi realizada uma revisão da literatura de forma a enquadrar e perceber a importância da temática no setor bancário. Verificamos que este setor é extremamente concorrencial e uma das estratégias utilizadas pelos bancos é a utilização de plataformas informáticas capazes de disponibilizar serviços online e por outro lado permitirem a extração de informação (por exemplo: padrões de comportamento dos seus clientes) dos dados recolhidos. A capacidade de extração de informação de grandes quantidades de dados, permite às instituições adaptar e melhorar o seu leque de produtos e obter o maior conhecimento possível sobre o cliente, tornando possível a antecipação de eventuais necessidades e corresponder com determinados produtos financeiros.

6.1. Reflexão sobre o contributo do estágio para o desenvolvimento de competências

A realização do estágio demonstrou-se bastante benéfica para o desenvolvimento de competências, uma vez que permitiu explorar algumas das áreas integrantes do mestrado: modelação, bases de dados e programação. Além destas *hard skills*, o estágio permitiu também desenvolver e melhorar *soft skills* essenciais para a vida profissional: trabalho em equipa, criatividade na resolução de problemas, responsabilidade, negociação, e

capacidade de comunicação. Como resultado do trabalho desenvolvido no estágio, foi proposto um contrato de trabalho (SI Developer Consultant).

O apoio dado pela equipa de trabalho e pelos orientadores de estágio, permitiu que situações adversas fossem ultrapassadas. A integração foi fácil e o trabalho de equipa proporcionou melhorias na relação e comunicação existente no dia-a-dia. Por sua vez, a relação com os clientes, permitiu aplicar e desenvolver novas competências interpessoais e de comunicação.

Como pontos positivos do estágio, realçamos:

- Oportunidade de poder participar em projetos internacionais de grande dimensão e com equipas multidisciplinares;
- Integração e apoio da equipa de trabalho;
- Utilização de tecnologias e ferramentas a nível empresarial;
- Responsabilidade pelo suporte de uma plataforma.

Não foram relevantes as adversidades encontradas, uma vez que pude contar com o apoio da equipa de suporte, não deixando de concluir qualquer tarefa por força de algum constrangimento, encontrando sempre formas de ultrapassar essas mesmas barreiras e realizar o trabalho proposto. No entanto, convém realçar:

- a) Apesar de bastante bem-recebido e apoiado por todos os colegas de equipa, o facto de não estar acostumado às tarefas que iria desempenhar e a falta de autonomia para as desempenhar levou a que os primeiros dias tenham sido menos bons. Não no aspeto de ambiente proporcionado pela empresa, mas sim pelo pensamento que ainda não podia ajudar a equipa da melhor forma.
- b) Como o estágio foi realizado na área de suporte, o conjunto de tecnologias e ferramentas que é necessário dominar torna-se mais abrangente.
- c) Por sua vez, é dado suporte a várias plataformas com características e diferentes parametrizações.
- d) Outra dificuldade surge quando é necessário alterar código desenvolvido por outros programadores. O período necessário para compreender a lógica e o estilo de programação poderá ser demorado.

6.2. Considerações finais

Em suma, pensamos que todas as tarefas foram cumpridas com sucesso, tendo sempre levado a cabo uma postura de compromisso e responsabilidade. Os objetivos propostos para o presente estágio podem considerar-se cumpridos, tendo para tal sido necessário dominar e utilizar diversos conceitos adquiridos ao longo do percurso académico. Por sua vez, foi possível utilizar metodologias e técnicas conhecidas teoricamente. Através do trabalho em equipa, e entre equipas, foi possível experienciar e consolidar um conjunto de normas importantes para o sucesso de qualquer equipa num projeto de software.

Pode assim concluir-se, que toda a experiência vivida ao longo do presente estágio permitiu obter competências (*hard e soft skills*) pessoais e profissionais relevantes, tendo fornecido as bases necessárias para iniciar o futuro profissional nesta área.

Bibliografia

- Brechner, E. (2015). *Agile Project Management with Kanban*. Microsoft Press.
- Campbell, A. (2021). *Agile Leadership: Secret Practices and Tips for Successful Agile Leader*. Oxford.
- Deloitte. (2019). *The Deloitte Global Millennial Survey 2019*. Deloitte.
- Deloitte. (2020). *Blockchain for Financial Leaders: Opportunity vs. Reality*. Deloitte.
- Frain, B. (2020). *Responsive Web Design with HTML5 and CSS Develop future-proof responsive websites using the latest HTML5 and CSS techniques*. Packt Publishing; 3rd edition.
- Ganguly, A. (2015). Optimization of It and Digital Transformation. *European Journal of Business and Innovation Research*, pp. 1-13.
- Gopane, T. J. (2018). The Likelihood of Financial Inclusion in e-Banking: A BiProbit Sample-Selection Modeling Approach. Em T. J. Gopane, *Digital Economy: Emerging Technologies and Business Innovation* (p. 205). Springer.
- INE. (2019). *Boletim Mensal de estatística*. Instituto Nacional de Estatísticas.
- Jihene Ouakdi, D. G. (2018). Innovation, Banking Development and Governance. *Third International Conference, ICDEc 2018* (pp. 79-94). Springer.
- Lawrence, E. (2015). *Debugging with Fiddler: The complete reference from the creator of the Fiddler Web Debugger*. Material.
- Microsoft. (2020). *.NET documentation*. Obtido de .NET documentation: <https://docs.microsoft.com/en-us/dotnet/>
- Motiwalla, M. A. (2019). Uncovering unobserved heterogeneity bias: Measuring mobile banking system success. *International Journal of Information Management*, pp. 439-451.
- Nagel, C. (2021). *Professional C and .NET*. Wrox.
- Raymond, A. R. (2018). Consumer Behaviour Theory: A Theoretical and Practical Approach in Banking Industry. *International Journal of Ayurveda*, pp. 26-34.
- Reis, A. R. (2017). *A utilização do mobile banking em alternativa ao atendimento presencial*. ULHT.
- Schindler, J. W. (2019). *Fintech and financial innovation: drivers and depth*. Finance and. Springer.
- Silva, V. P. (2003). Competitividade em tempo de mudança: uma banca diferente surgiu nas décadas de 80 e 90. *Revista*, pp. 5-56.
- Sripalawat, M. T. (2011). M-banking in metropolitan Bangkok and a comparison with other countries. *Journal of computer information systems*, pp. 67-76.

- Stojanović, V. T. (2019). Trends and Innovations in Mobile Banking. Em *Digital transformation: new challenges and business opportunities*. (pp. 290-310). Silver and Smith Publishers,.
- Umbraco. (2022). *Documentation Umbraco - Getting Started* . Obtido de Documentation Umbraco - Getting Started : <https://our.umbraco.com/documentation/Getting-Started/>
- Zhou, T. (2018). Examining users' switch from online banking to mobile banking. *International Journal Networking and Virtual Organizations*, pp. 18-28.

