



Bernardo Antunes  
Gomes Augusto

## **Desenvolvimento de Dashboard em Power BI**

Relatório de Estágio  
Licenciatura em Bioinformática

### **ORIENTADOR**

Prof. Doutor Vítor Barbosa

### **SUPERVISOR**

Anderson Abreu

Julho 2020

## Agradecimentos

Este relatório resume o caminho percorrido, esforço e dedicação prestados ao longo destes 3 anos de licenciatura de Bioinformática. Todas as palavras de incentivo que me foram transmitidas, todos os meus altos e baixos e todas as conquistas.

Agradeço ao Professor Pedro Manuel Garcia Vieira Neto enquanto Diretor da Escola Superior de Tecnologias do Barreiro pertencente ao Instituto Politécnico de Setúbal.

Ao meu orientador, Professor Doutor Vítor Barbosa, agradeço todo o apoio por ter sido um elemento fundamental nesta jornada.

À empresa *Integer Consulting*, ao CEO, Luís Setúbal, ao meu supervisor, Anderson Abreu, e a todos os meus colegas de trabalho, agradeço a forma como me acolheram nesta grande equipa e me transmitiram imenso conhecimento e prática que adquiriram ao longo dos anos, fazendo com que pudesse aprofundar o meu conhecimento nesta área.

Um especial agradecimento a dois professores que foram cruciais para o meu sucesso, o professor Francisco Pina Martins, que me ajudou imenso a ultrapassar as imensas dificuldades no meu primeiro ano de licenciatura e que influenciou de forma extrema o meu crescimento como profissional e pessoa. E deixar também um especial obrigado à professora Raquel Barreira, que me mostrou especial preocupação em aconselhar e acalmar, as preocupações de estágio, tanto minhas como dos meus colegas.

Por fim, gostaria de agradecer à minha família e amigos, que se mostraram sempre presentes, para me aconselhar, para partilhar a carga, e para me motivarem a ir sempre mais além e alcançar tudo o que sempre quis. Para sempre vos estarei grato.

# Índice

Agradecimentos .....	ii
Índice .....	iii
Índice de Figuras.....	iv
Índice de Tabelas.....	v
Lista de Siglas e Abreviaturas .....	vi
Resumo.....	vii
Abstract.....	1
Introdução .....	2
1   Objetivos do estágio .....	3
2   Caracterização da Empresa .....	4
3   Atividades desenvolvidas .....	5
3.1   Cronograma do estágio.....	6
4   Fundamentos teóricos .....	7
4.1   Business Intelligence .....	7
4.2   ETL.....	9
4.3   Data Warehousing .....	10
4.3.1   Surrogate Key x Natural Key.....	12
5   Procedimentos/Metodologias/ Material e Métodos.....	14
5.1   Power BI .....	14
5.2   SQL Server Management Studio .....	15
5.3   Metodologia Scrum .....	15
5.4   Metodologia de base de dados .....	17
5.5   Metodologia de BI .....	17
6   Apresentação e Discussão das Atividades .....	19
7   Conclusões.....	33
8   Conclusions.....	34
Referências.....	35

# Índice de Figuras

<i>Figura 1 - Passos BI .....</i>	<i>8</i>
<i>Figura 2 - Estrutura SCRUM .....</i>	<i>16</i>
<i>Figura 3 - Procedimento BI.....</i>	<i>18</i>
<i>Figura 4 - Organização das atividades.....</i>	<i>19</i>
<i>Figura 5 - Esquema ETL .....</i>	<i>20</i>
<i>Figura 6 – Tabela dim_candidate .....</i>	<i>21</i>
<i>Figura 7 - Índice surrogate key candidate .....</i>	<i>21</i>
<i>Figura 8 - Procedure candidate.....</i>	<i>22</i>
<i>Figura 9 - Bloco de código ET .....</i>	<i>23</i>
<i>Figura 10 - Bloco de código L .....</i>	<i>23</i>
<i>Figura 11 - Página de Contactos.....</i>	<i>24</i>
<i>Figura 12 - Candidatos distintos.....</i>	<i>24</i>
<i>Figura 13 - Total contactos.....</i>	<i>25</i>
<i>Figura 14- Tooltip Contactos x Tipo de Contacto.....</i>	<i>25</i>
<i>Figura 15 - DAX Contactos x Tipo de Contacto .....</i>	<i>26</i>
<i>Figura 16 - Página de Pedidos.....</i>	<i>26</i>
<i>Figura 17 - Total de pedidos .....</i>	<i>27</i>
<i>Figura 18 - Tooltip top 10 clientes (pedidos).....</i>	<i>28</i>
<i>Figura 19 - Página CVs x Entrevistas .....</i>	<i>28</i>
<i>Figura 20 - measure totais.....</i>	<i>29</i>
<i>Figura 21 - Selecionar filtro .....</i>	<i>29</i>
<i>Figura 22 - Selecionar tipo de visualização.....</i>	<i>30</i>
<i>Figura 23 - Selecionar filtro para o gráfico .....</i>	<i>30</i>
<i>Figura 24 - Measure adjudicados no mês passado .....</i>	<i>31</i>
<i>Figura 25 - Measure adjudicado no mês atual.....</i>	<i>31</i>
<i>Figura 26 - Measure comparativo com adjudicados mês passado.....</i>	<i>31</i>
<i>Figura 27 - Seleção do gráfico e campos necessários .....</i>	<i>32</i>

# Índice de Tabelas

<i>Tabela 1 - Cronograma do estágio.....</i>	<i>6</i>
--	----------

## **Lista de Siglas e Abreviaturas**

*BI – Business Intelligence*

*DAX - Data analysis expressions*

*DW - Data Warehousing*

*ETL – Extract Transform and Load*

*OLAP – Online Analytical Processing*

*Sk – Surrogate Key*

*SQL - Structured Query Language*

*SSMS – SQL Server Management Studio*

## Resumo

Neste relatório é apresentado o trabalho desenvolvido ao longo do estágio curricular, no âmbito da licenciatura em Bioinformática, o qual foi executado com o recurso a uma ferramenta de *business intelligence*, um sistema de gestão de bases de dados e a sua aplicação de gestão, *Power BI*, *SQL Server* e *SQL Server Management Studio*, respetivamente.

O tema de “Business Intelligence” é o elemento central deste relatório e, como tal, será apresentado um enquadramento de *business intelligence* e outros conceitos complementares. Para além dos conceitos teóricos de enquadramento, será dada especial atenção às tecnologias e metodologias utilizadas e como estas foram relevantes para atingir os objetivos do estágio.

As atividades desenvolvidas ao longo do estágio são demonstradas e explicadas com alguns exemplos.

O trabalho desenvolvido no estágio e apresentado neste relatório permite demonstrar de que forma a *Business Intelligence* pode contribuir para um incremento na produtividade da gestão de topo e auxiliar na melhor tomada de decisão, com base em dados.

## Palavras Chave

*Business Intelligence*; Base de dados; *SQL*; *DAX*;





## **Abstract**

*This report presents the work developed during the curricular internship, within the scope of the degree in Bioinformatics, which was carried out using a Business Intelligence tool, a database management system and its management application, Power BI, SQL Server and SQL Server Management Studio, respectively.*

*The theme of “Business Intelligence” will be the center of this analysis, and such, a framework of Business Intelligence and other complementary concepts will be presented. In addition to theoretical framework concepts, special attention will be given to the technologies and methodologies used and how they were relevant to achieve the internship’s objectives.*

*The activities developed during the internship are demonstrated and explained with some examples.*

*The work developed in the internship and presented in this report allows to demonstrate how Business Intelligence can contribute to an increase in top management productivity and help in better decision-making, based on data.*

## **Keywords**

*Business Intelligence; Databases; SQL; DAX;*

## Introdução

Através da realização do estágio apresentado neste relatório, foram utilizadas algumas das aprendizagens, conhecimentos e competências obtidas ao longo da Licenciatura em Bioinformática. Estas, em conjunto com o apoio dos colegas de trabalho da Integer Consulting, permitiram uma integração na área de Business Intelligence com um maior nível de conforto.

Infelizmente, devido à pandemia de Covid-19 e consequentes restrições, foi necessário realizar o estágio na modalidade de trabalho remoto. Mas tal nunca se demonstrou um obstáculo, porque tanto o supervisor de estágio como os colegas da Integer Consulting sempre estiveram presentes para auxiliar em qualquer coisa que necessitasse.

As funções desempenhadas durante o estágio consistiram em aprofundar os conhecimentos de SQL e Power BI, tal como aprender a metodologia SCRUM e a utilizar a ferramenta DevOps Azure para organização das tarefas do projeto. Estas tarefas foram acompanhadas de formação, sugerida pelo meu supervisor, e algumas por iniciativa própria. O objetivo principal consistiu na criação de um sistema de indicadores com o auxílio de ferramentas mencionadas anteriormente que auxiliassem os gestores da Integer Consulting no processo de tomada de decisão. A criação destes indicadores foi proposta devido à necessidade de análise na gestão do ciclo de entrevista dos candidatos e na gestão de pedidos de profissionais pelos seus clientes.

A oportunidade de ingressar no mercado de trabalho numa empresa como a Integer Consulting abre novas possibilidades numa área que sempre se mostrou interessante e contribuiu tanto para o desenvolvimento das capacidades em tecnologias já introduzidas durante a licenciatura, assim como em novas cujo primeiro contacto ocorreu durante o estágio.

# 1 Objetivos do estágio

Este estágio teve como objetivo desenvolver conhecimentos e capacidades no que toca a utilização de técnicas de BI e técnicas de análise de negócio.

Estas técnicas serviram para a criação de um conjunto de indicadores como, Colaborador x pedidos/propostas para clientes, Colaborador x propostas/adjudicações em clientes, Quantidade de contatos com candidatos, entre outros. Estes indicadores pretendem auxiliar os gestores da Integer na tomada de decisão. Esses indicadores serão utilizados para obter dados que responderão a questões atuais e futuras. Ao comparar os resultados desejados com os dados detetados, a gestão da Integer espera canalizar as energias, foco e conhecimentos específicos para alcançar as metas de curto, médio e longo prazo.

Neste sentido, a criação e execução do projeto de indicadores implica na aplicação prática de *design patterns*, fórmulas estatísticas, técnicas de *Business Intelligence* e técnicas de análise de negócio. Assim, a extração de dados resultará em tabelas e gráficos que agrupados em conjunto serão disponibilizados em *dashboards*.

Para além das componentes mais técnicas (BI e técnicas de análise de negócio) também era um objetivo adquirir uma postura profissional proativa e crítica para poder trabalhar em equipa ou individualmente. Esta componente menos técnica também pôde ser alcançada durante o estágio realizado através da contínua melhoria dos indicadores desenvolvidos, acrescentando tanto funcionalidades que melhorassem a *user experience* como em termos de design gráfico.

As tecnologias em destaque estão expostas com maior detalhe na secção que será brevemente apresentada denominada por “Atividades Desenvolvidas”.

## 2 Caracterização da Empresa

A Integer Consulting é uma empresa portuguesa de consultoria em tecnologias de informação. Tem como foco a área de IT outsourcing, software, projetos de *nearshore* e desenvolvimento à medida.

A Integer é parceira dos principais players tecnológicos a nível mundial e trabalha com empresas nacionais e multinacionais de diversos setores de atividade. A maioria dos projetos em que participamos são em Portugal, no Brasil e um pouco por toda a Europa. O talento tecnológico e a experiência das pessoas que fazem parte das equipas e que integram os projetos é algo que diferencia a Integer de outras empresas. A empresa possui escritórios em Lisboa e no Porto. Em Lisboa o escritório situa-se na R. Julieta Ferrão, 1600-131.

A Integer Consulting foi criada no início de 2007 com a visão de que o espírito de equipa e o trabalho íntegro são a base do sucesso de uma empresa. Esta ideologia do CEO, Luís Setúbal, transformou a Integer naquilo que é hoje, uma empresa de referência reconhecida pelos profissionais de IT, um fornecedor de longa data no setor tecnológico para muitas empresas de renome e um parceiro que honra compromissos.

A Integer Consulting é uma empresa que tem a ambição de continuar a crescer, mas sem nunca esquecer os seus quatro valores chave. Estes são a integridade, parceria, compromisso e trabalho de equipa.

### 3 Atividades desenvolvidas

Durante o período de estágio foi possível realizar todas as tarefas que foram inicialmente propostas e completar o projeto de estágio sem complicações relevantes. Durante quase todo o estágio foram realizadas atividades acompanhadas de formação em *Power BI*, *SQL* e fundamentos de *Data Warehousing*.

Numa fase inicial foi necessária a consulta dos documentos de metodologia de base de dados e de BI. Estes documentos foram essenciais para diversas tarefas. O documento de Metodologia de Base de Dados foi útil para a criação de tabelas e colunas e a nomenclatura das mesmas. O documento de Metodologia de BI foi útil para perceber como se deve proceder num projeto de BI.

Seguidamente avançou-se para o levantamento de requisitos, com os gestores da *Integer Consulting*, para identificar quais os indicadores que seriam tratados naquele momento e que tabelas da base de dados seria necessário consultar para obter os dados. Nesta tarefa criou-se e atualizou-se a documentação do sistema de indicadores.

Após o levantamento de requisitos foi necessário o desenvolvimento de *stored procedures* e criação de novas tabelas para ser possível desenvolver os indicadores solicitados (processo de ETL). Estas *stored procedures* e tabelas foram criadas no *SQL Server*, uma ferramenta da Microsoft.

Por fim, a última atividade consistiu na definição da melhor forma de visualização dos dados (gráficos, tabelas, cartões, ...) e a construção desses componentes na ferramenta *Power BI* da Microsoft.

### 3.1 Cronograma do estágio

As atividades foram executadas de acordo com a seguinte calendarização.

*Tabela 1 - Cronograma do estágio*

	Meses				
Atividades	Março	Abril	Maio	Junho	Julho
<b>Aquisição de conhecimento teórico</b> (formação e aquisição de conhecimento de <i>Power BI</i> , <i>SQL</i> , fundamentos de <i>Data Warehousing</i> e análises preditivas)	X	X	X	X	
<b>Levantamento de Requisitos</b> (reuniões com os managers da Integer Consulting e desenvolvimento de documentação)	X	X	X	X	
<b>Processo de ETL</b> (criação de <i>stored procedures</i> e tabelas)	X	X	X	X	
<b>Sistema de Indicadores</b> (construção gráficos e/ou tabelas no <i>Power BI</i> )	X	X	X	X	
<b>Aplicação da metodologia de base de dados</b> (Renomear tabelas e colunas e corrigir <i>procedures</i> )				X	
<b>Relatório de Estágio</b>					X

## **4 Fundamentos teóricos**

Nesta secção serão incluídos conceitos teóricos uteis para a compreensão do porquê de as atividades desenvolvidas durante o estágio terem sido efetuadas de uma certa forma e não de outra. Estes são tanto conceitos já abordados ao longo da licenciatura em Bioinformática, assim como novos adquiridos durante este período de estágio.

### **4.1 Business Intelligence**

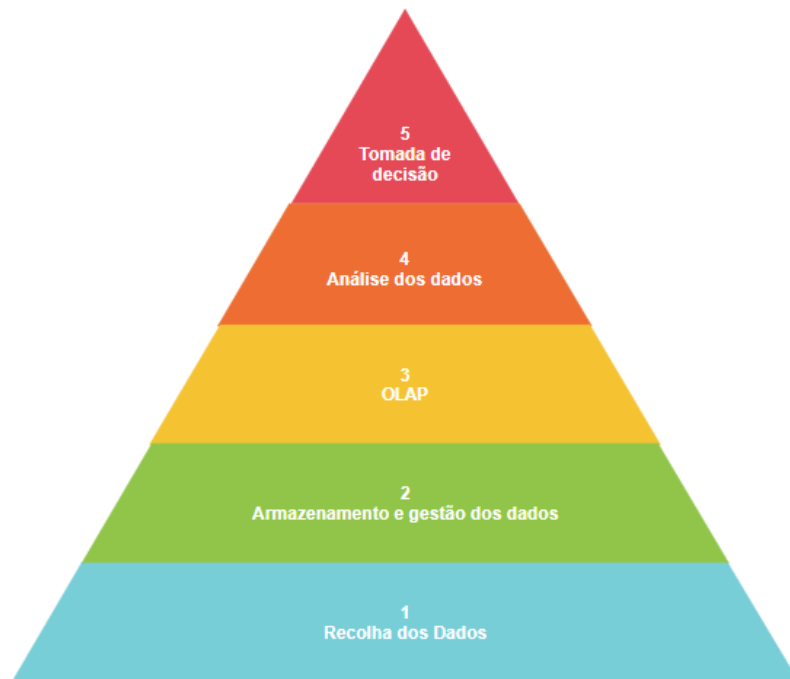
BI é a transformação de dados em informação, com a finalidade de termos uma gestão do negócio mais eficiente, com o auxílio de técnicas e ferramentas [1]. É utilizado para a tomada de decisão. Podem ser decisões básicas de operações ou ser decisões estratégicas de negócio.

O objetivo de BI é permitir uma fácil leitura e interpretação do grande volume de dados da empresa/organização. Esta interpretação dos dados é mais eficaz quando combinada a dados externos de mercado, por exemplo, comparar como se encontram as vendas do produto face aos concorrentes.

A grande quantidade de dados ajuda a identificar e a criar estratégias de negócio. Levando à identificação de novas oportunidades e implementação de estratégias fomentadas em dados recolhidos. Isto vai gerar vantagem competitiva a longo prazo.

BI está estruturado em vários passos (Figura 1). Neste relatório apenas vão ser abordados os seguintes: extração dos dados, armazenamento e gestão dos dados, visualização e análise dos dados.

A recolha dos dados é o primeiro passo para os poder analisar. Para entender e resolver os problemas de negócio, os analistas de dados, devem ter uma boa compreensão dos dados com que estão a trabalhar. Questões como: “Como se extrai os dados?”, “Que tipo de dados existem?”, “De onde vem estes dados?”. São tudo questões que os cientistas de dados devem saber as respostas antes de partir para análise dos dados [2]. Ao sabermos as respostas a estas questões construímos uma base à qual a nossa análise se vai reger.



*Figura 1 - Passos BI*

Armazenamento e gestão dos dados é um dos aspetos base de ciência de dados [2]. Dados são nada mais nada menos do que factos que podem ser armazenados e processados por computadores. Para manipular e processar os dados de forma eficiente, é muito importante que os mesmos sejam armazenados de forma apropriada. Os dados existem em diversas formas e tamanhos e algumas das formas mais comuns de dados são números, texto, imagens e vídeo. Dependendo do tipo de dados, existem diversas formas para o seu armazenamento e processamento. Uma das mais conhecidas de armazenamento dos dados é o armazenamento em bases de dados relacionais. As bases de dados relacionais regem-se por dois princípios: integridade de identidade e integridade de referencial. Na integridade de identidade todos os dados são organizados por um valor único chamado chave primária. Na integridade referencial diz-se que devemos ter uma relação consistente, por exemplo, a coluna de chave estrangeira deve ser igual à coluna de chave primária da outra tabela. O modelo das bases de dados relacionais assenta no uso de tabelas para guardar dados.

A análise de dados é uma área enorme e em continua expansão com métodos a emergir todos os dias para poder dar explorar e dar sentido à enorme quantidade de dados que existe. No entanto, com qualquer conjunto de dados ou resultado de



análise, a principal preocupação está em comunicar os resultados ao leitor. Infelizmente a interpretação humana não está otimizada para entender a relação entre grandes (ou mesmo moderados) conjuntos de dados. No entanto a percepção humana é excelente para entender inter-relações entre conjuntos de dados, como desvios, series e semelhanças através de representações visuais [2]. Por essa razão a visualização dos dados constitui um papel fundamental em BI.

## 4.2 ETL

ETL é um tipo de *data integration* em três etapas (extração, transformação, carregamento) usado para combinar dados de diversas fontes [3]. O ETL consome uma quantidade significativa do tempo e esforço na construção de um ambiente DW/BI. Em desenvolvimento, o sistema ETL é desafiador devido à quantidade de restrições externas que leva ao aumento da pressão sobre o design do mesmo, por exemplo, nos requisitos de negócio, orçamento e elementos da equipa que se encontram disponíveis [4].

No entanto pode ser difícil compreender o porquê de o ETL ser um sistema tão complexo e consumir tantos recursos. O significado das três letras é de fácil compreensão: obter os dados do local de origem (**E**xtract), faz-se algo com os dados (**T**ransform) e por fim deposita-se os dados (**L**oad) num conjunto de tabelas para os utilizadores finais consultarem.

Quando se questiona a melhor maneira de planear e construir um sistema de ETL, a resposta mais comum é: “Isso depende”. Depende da fonte, das limitações dos dados, das linguagens de programação, das ferramentas de ETL disponíveis, dos conhecimentos da equipa e das ferramentas de BI. Porém a resposta “depende” é perigosa pois pode levar à adoção de uma abordagem não estruturada para desenvolver um sistema de ETL, que na pior das hipóteses resulta numa “*spaghetti-mess*” que vai levar à existência de tabelas, *procedures*, *triggers* e *jobs* indiferenciadas. Esta abordagem de “design” criativo não deve ser tolerada [4].

Com a sabedoria de retrospectiva de milhares de DWs de sucesso, um conjunto de melhores praticas de ETL foram estabelecidas. Não há razão para se tolerar uma solução não estruturada.

### 4.3 Data Warehousing

Um DW é utilizado para armazenar informações relativas às atividades de uma organização em bancos de dados, de forma consolidada. Um DW é composto por tabelas de dimensão e de facto. As tabelas de dimensão são normalmente tabelas desnormalizadas, com muitas colunas e com muitos atributos de texto de baixa cardinalidade [5]. As tabelas de dimensão são o alvo principal das restrições e especificações de queries e aplicações de BI. As tabelas de facto são inteiramente baseadas em atividade física e não são influenciadas por relatórios que possam surgir. A tabela de factos é composta por medidas numéricas e contém chave estrangeira para cada uma das tabelas de dimensão associadas [6]. As tabelas de facto são o principal alvo de cálculos e agregações dinâmicas que surgem em queries. Quanto a tabelas de facto podemos ter accumulating snapshot ou periodic snapshot. Uma linha numa accumulating snapshot resume muitos eventos de medição, que ocorreram num período padrão como um dia, uma semana ou um mês [7]. Numa periodic snapshot temos que uma linha resume os eventos de medição que ocorrem em passos previsíveis entre o início e o fim de um processo [8].

Um DW deve tornar fácil o acesso às informações. O conteúdo do sistema de DW/ BI deve ser compreensível. Os dados devem ser intuitivos e óbvios para o utilizador final, não apenas para o desenvolvedor. As estruturas de dados devem coincidir com a o raciocínio de negócio e linguagem usada pelos utilizadores. Os utilizadores finais desejam sempre separar e combinar dados analíticos em imensas combinações. As ferramentas e aplicações de BI que acedem aos dados devem ser simples e fáceis de usar. Estes devem retornar os resultados de consulta ao utilizador no menor tempo possível. Ou seja, o DW tem de ser rápido e simples.

Outra característica que o DW deve possuir é que deve mostrar informação de forma consistente. Os dados no sistema de DW/ BI deve ser confiável. Os dados devem ser escolhidos cuidadosamente de diversas fontes de dados, limpos, de qualidade garantida e apenas disponibilizados quando adequados para consumo pelo utilizador. A consistência também implica rótulos e definições comuns para o conteúdo do sistema de DW/ BI. Se duas medidas de desempenho tem o mesmo nome devem significar a mesma coisa. Se não significarem, devem possuir nomes diferentes.

O DW deve adaptar-se a mudanças. Necessidades do utilizador, condições do negócio, dados e tecnologia estão sujeitos a alterações. O sistema DW/ BI deve ser projetado para lidar com mudança inevitável de forma harmoniosa, de modo a que não invalide os dados ou aplicações existentes. Dados e aplicações existentes não devem ser alterados ou interrompidos quando novos dados são adicionados ao DW. Se os dados descritos devem ser modificados, deve-se avisar o utilizador acerca das mudanças e ser o mais transparente possível.

O DW deve apresentar informações em tempo útil. DW/ sistemas de BI são utilizados intensamente para decisões operacionais e por isso os dados brutos precisam ser convertidos em informações acessíveis no menor tempo possível. A equipa de DW/ BI e os utilizadores precisam de estabelecer expectativas sobre o que significa fornecer dados quando há pouco tempo para limpeza ou validação.

O sistema de DW/ BI deve ser seguro. As informações mais importantes da organização são guardadas no DW. No mínimo, o DW contém informações do que o que é que a empresa vende, a quem, e o preço que pratica. Detalhes que podem ser prejudiciais nas mãos erradas. O sistema de DW/ BI deve controlar o acesso às informações confidenciais da organização.

Deve conter dados confiáveis. O sistema deve servir de base como base autorizada e de confiança para uma melhor tomada de decisão. O DW deve ter os dados corretos para apoiar o processo de tomada de decisão. Os resultados mais importantes de um sistema de DW/ BI são as decisões tomadas com base na evidência analítica. Estas decisões geram impacto no negócio e valor para a empresa e para o sistema.

Por último, a comunidade empresarial deve aceitar o sistema para ser considerado bem-sucedido. Não interessa o quão bom o sistema é. Se a comunidade empresarial não o utiliza, o projeto foi um falhanço. Ao contrário de um sistema operacional onde o utilizador não tem escolha se não o usar, aqui tem essa escolha. O sistema vai ser aceite se for visto como uma fonte simples e rápida para aceder a informações relevantes para o negócio que pode levar o mesmo ao sucesso.

#### 4.3.1 Surrogate Key x Natural Key

Uma questão que é sempre colocada é se, se deve utilizar uma *natural key* ou uma *surrogate key* (sk) como chave primária. A sk é um valor gerado pelo sistema sem nenhum significado para o negócio. Esta é usada estritamente apenas para identificar um registo em uma tabela. Quanto à *natural key* é uma coluna ou conjunto de colunas que já existem na tabela. Esta tem um significado comercial.

O uso da *natural key* como chave primaria traz vantagens, mas também desvantagens.

Quanto a vantagens temos que as chaves têm significado para o negócio e pode ser usada como campo de procura quando se faz uma *query* na base de dados. Temos também que como estas colunas já existem não vai ocupar mais espaço no disco.

Mas o uso de *natural keys* como chave primaria também traz desvantagens. Uma delas é estas suscetíveis a mudar. Se os requisitos de negócios mudarem esta também terá de mudar. Por exemplo se foi usado o número de segurança social para o funcionário e a empresa expande para fora do país, nem todos os funcionários vão ter um número de segurança social no mesmo formato. E teria de ser criada uma nova chave. Esta é também mais difícil de manter se depender de várias colunas. Pior performance também é uma das desvantagens [9]. Visto que esta coluna tem valores maiores ou é feita a partir de outras colunas. E não podem ser introduzidos novos valores até este valor ser introduzido.

Quanto a vantagens no uso de sk como chave primária, esta não tem qualquer valor para o negócio. Usando o exemplo acima do número da segurança social. Não teríamos de arranjar uma nova chave primária, pois esta não tem valor comercial. Poderíamos adicionar funcionários de outro país que tem um nome diferente para o número de segurança social através da criação de uma nova coluna. É necessário menos código para manter esta chave. Isto porque normalmente é um número sequencial e que incrementa quando um registo novo é feito. Melhor performance também é uma das vantagens. Isto porque os valores da chave vão ser menores, ocupando assim menos espaço no disco [9].

Quanto a desvantagens acerca do uso de sk temos que é uma nova coluna que tem de ser criada, ocupando mais espaço no disco. São necessário mais *joins* a tabelas filhas, visto que os valores não têm sentido para o negócio por si só [9]. É também mais difícil de distinguir dados de teste de dados de produção. Por exemplo valores da sk são apenas gerados automaticamente sem significado comercial. Tornando-se assim difícil concluir se alguém introduziu dados de teste na base de dados de produção.

## 5 Procedimentos/Metodologias/ Material e Métodos

Nesta secção serão incluídos os procedimentos, metodologias, métodos e materiais utilizados durante este estágio curricular. Estes são tanto ferramentas já utilizadas e exploradas ao longo da licenciatura em Bioinformática, assim como novas adquiridas durante este período de estágio.

### 5.1 Power BI

O Power BI é um serviço de BI da Microsoft lançado a 24 de julho de 2015. O objetivo do Power BI é fornecer visualizações interativas e recursos de BI com uma interface simples para que os utilizadores finais criem os seus próprios relatórios e dashboards.

Os utilizadores de Power BI podem para além de criar os seus próprios relatórios, conectar a base de dados, transformar e modelar dados e compartilhar os relatórios com outras pessoas utilizando o serviço de Power BI.

Power Bi é utilizado por várias razões, mas algumas das quais são o seu arranque rápido [10]. Pode-se obter *insights* muito rapidamente sem ser necessário muita prática.

Outra razão é o facto de a publicação e distribuição ser muito simples [10]. Em vez de enviar arquivos grande por e-mail ou colocando em discos externos pode-se enviar diretamente para o Power Bi Web e apenas partilhar o link do dashboard e toda a gente com o link pode aceder ao dashboard.

O facto de termos dados em tempo real também é extremamente útil pois assim temos sempre os dados atualizados [10]. Isto dá a capacidade ao utilizador de resolver problemas e identificar oportunidades rapidamente.

Também temos a capacidade de personalizar a *user experience* podendo assim torna-la a melhor possível para podermos ter uma grande aceitação do dashboard por parte da audiência.

Por último, temos a capacidade de personalizar os recursos de segurança [10]. Permitindo assim que os gestores de topo tenham acesso a toda a informação e os

utilizadores que não tem necessidade de aceder a toda a informação, apenas acedem aquela a que lhes afeta ou faz sentido verem.

## 5.2 SQL Server Management Studio

O SQL Server Management Studio (SSMS) é um ambiente para gestão de qualquer infraestrutura SQL, do SQL Server à Base de Dados SQL do Azure. SSMS utiliza SQL (Structured Query Language) para pesquisa declarativa em bases de dados relacionais [11].

SQL é inspirado em álgebra relacional. O SSMS permite configurar, monitorizar e administrar instâncias do SQL Server e base de dados. Esta ferramenta é utilizada para implementar, monitorizar e atualizar componentes da camada de dados usadas por aplicações e criar consultas e scripts.

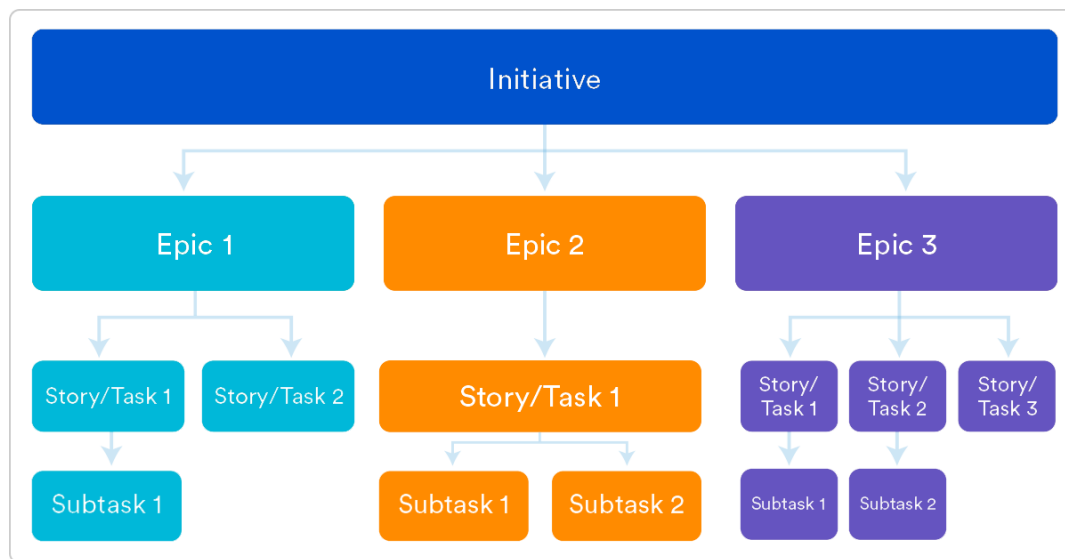
## 5.3 Metodologia Scrum

Scrum é uma metodologia de desenvolvimento ágil utilizada no desenvolvimento de software baseada em processos iterativos e incrementais. SCRUM é uma framework ágil, adaptável, rápida, flexível e eficaz, projetada para entregar valor ao cliente durante o desenvolvimento do projeto [12]. O objetivo principal do SCRUM é a satisfação das necessidades do cliente através de um meio de transparência na comunicação, responsabilidade coletiva e progresso contínuo. O desenvolvimento parte de uma ideia geral do que precisa ser construído, elaborando-se uma lista de características ordenadas por prioridade (*product backlog*) que o utilizador final deseja obter.

A organização do projeto é feita através de *user stories* e *tasks*. Uma *user story* é uma explicação geral de um recurso de software escrita a partir da perspetiva do utilizador final. Uma *task* é criada dentro de uma história e normalmente refere-se a uma melhoria.

User stories são peças fundamentais para a construção de estruturas maiores como, epics ou iniciativas [13]. Epics são grandes itens de trabalho divididos em várias histórias que juntos formam a *iniciative*. Estas estruturas de maior dimensão contribuem para que o trabalho da equipa do dia a dia influencie objetivos

organizacionais. Podemos observar como se organizam as estruturas faladas anteriormente na figura 2.



*Figura 2 - Estrutura SCRUM*

Existe uma repetição sequencial de reuniões realizadas antes, durante e depois de cada Sprint, em projetos Scrum. Uma Sprint representa o ciclo de trabalho, neste caso trata-se de duas semanas.

A metodologia Scrum possui 4 cerimónias: a Sprint Planning Meeting, Daily Scrum, Sprint Review Meeting e a Sprint Retrospective Meeting [14].

Na Sprint Planning Meeting é onde se planeia o trabalho que irá decorrer na Sprint e é necessário que toda a equipa esteja presente [15].

A Daily Scrum é um evento diário e cronometrado onde é feito um ponto de situação das próximas 24 horas [16]. Neste caso a Daily teve uma duração de 15 minutos no máximo.

A Sprint Review Meeting pretende analisar a lista de tarefas do projeto e assim perceber qual o valor acrescentado ao produto em que a equipa está a trabalhar [17].

Na cerimónia de Sprint Retrospective Meeting a equipa deve reunir para refletir sobre a Sprint que terminou para que consigam perceber o que pode ser melhorado. A Retrospective permite que a equipa se concentre no seu desempenho geral e identifique estratégias de melhoria contínua [18].



## 5.4 Metodologia de base de dados

Como mencionado na introdução uma das metodologias que existe na Integer Consulting é a metodologia de base de dados. Esta metodologia tem como objetivo padronizar elementos da base de dados de acordo com as orientações ISO/IEC 11179-5. ISO/IEC 11179-5 é um padrão internacional que serve para a padronização e registo de meta dados com o intuito de torna-los dados compreensíveis e partilháveis.

Este documento foi usado maioritariamente para a nomenclatura das tabelas, colunas e *procedures* criadas.

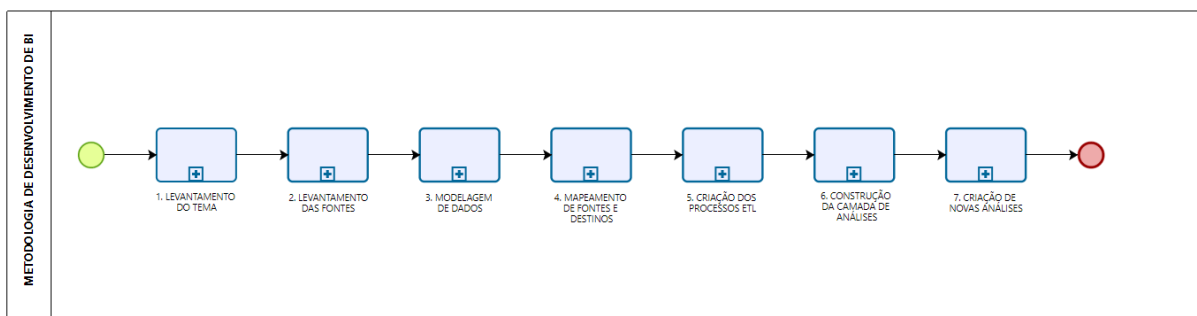
Em termos de nomenclatura de *procedure* temos que vai ser deste género **spn\_nomeprocedimento**. Assim com para darmos *load* aos dados na tabela *dim\_candidate* a *procedure* irá ter o nome, *spn\_load\_dim\_candidate* [19].

No caso de tabelas, caso sejam tabelas de dimensão terão o prefixo **dim**. Se, se tratarem de uma tabela de facto terão **fact**. Ambas seguidas de um **\_** antes do nome da dita tabela.

Em termos de colunas. Caso se trate de uma surrogate key teremos o prefixo **sk** seguido do nome da tabela (não incluído o tipo de tabela para não ser um nome muito extenso). Se se tratar de um código teremos o prefixo **cd** antes do nome do código em si. E caso se trate de um nome ou descrição terá o prefixo **ds** seguido do nome da coluna.

## 5.5 Metodologia de BI

O outro documento de metodologia que a empresa possui é o documento de Metodologia de BI. Este documento tem descrito como um processo de BI se deve proceder e descrições dos seus passos. Na figura 3, temos um esquema resumo de como tudo isto se processa, retirado do documento de Metodologia de BI [20].



*Figura 3 - Procedimento BI*

Neste esquema temos as etapas numeradas e começando pela 1, Levantamento do Tema. Nesta etapa vai ser definido o tema/assunto do projeto de BI, informando a área de negócio responsável e apresentando a justificação para escolha do tema. Irão também ser criados os documentos de requisitos e de visão, que descrevem as exigências propostas pelo cliente.

Na etapa 2, Levantamento das fontes. Aqui terá de ser identificar onde se encontram os dados necessários e como poderemos adquiri-los. Nesta etapa irá ainda existir entrevistas com especialistas das fontes de dados para garantir que está tudo correto.

Na etapa 3, será a modelação dos dados, e onde iremos estruturar a área de staging. Aqui irá também ser definido as tabelas que devem ser de dimensão e as que devem de ser de facto.

Na etapa 4, aqui iremos elaborar as especificações do processo de ETL. Irá ser definido a origem e o destino dos dados.

Na etapa 5 iremos criar os processos de ETL que foram elaborados na etapa anterior. Nesta etapa poderão surgir ajustes ao processo de ETL caso nos testes o mesmo não esteja a funcionar corretamente.

A etapa 6 consiste em conectar o DW à ferramenta de visualização, neste caso Power BI. Convém verificar se a importação dos dados foi bem-sucedida.

Na etapa 7 consiste em criar métricas estipuladas no levantamento de requisitos. Estas métricas podem estar disponibilizadas em cartões, gráficos, tabelas, etc.

## 6 Apresentação e Discussão das Atividades

Neste capítulo, serão apresentados e discutidos os indicadores criados neste estágio. Muitos dos indicadores foram solicitados pelos gestores da Integer Consulting, mas existem alguns que foram ideias sugeridas pela equipa de BI da empresa.

Para obter os indicadores desejados foi necessária a criação de novas tabelas e *procedures*. Foi utilizado o SSMS para a construção do processo de ETL. Primeiramente foram construídas as tabelas onde seriam depois guardados os dados e ainda as respetivas *procedures*. Como exemplo será usada a tabela *dim\_candidate* a *procedure* correspondente, *spn\_load\_dim\_candidate*.

Antes de mais irá ser apresentado um esquema em que se poderá observar a organização das atividades desenvolvidas durante o estágio, figura 4. Antes de se começar com a construção dos gráficos e tabelas é necessário o levantamento de requisitos. O levantamento de requisitos consiste numa reunião com os managers para se perceber que tipo de métricas pretendem e para os mesmos nos explicarem onde podemos aceder aos dados para construir os gráficos ou tabelas solicitadas.

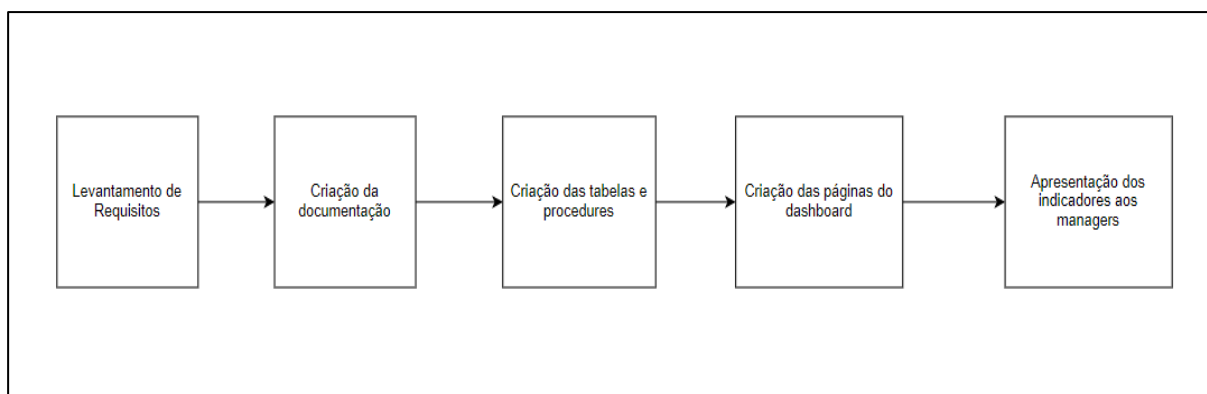


Figura 4 - Organização das atividades

Quando concluímos este levantamento de requisitos, passamos para a próxima fase que é a criação da documentação. A documentação consiste numa descrição dos indicadores solicitados e o que os mesmos significam. Nesta parte já é necessária a criação de uma história no DevOps Azure para questões de organização do projeto.

Avançando para a parte mais técnica do projeto, criação das tabelas e *procedures*. Nesta secção, tal como dito anteriormente, são criadas as tabelas e *procedures* necessárias para depois podermos construir os indicadores solicitados. Este é um processo de ETL que pode ser observado na figura 5. Basicamente são extraídos os dados de uma tabela de outra base de dados (*candidate*) e são depositados numa tabela temporária (*#tm\_candidate*). Uma tabela temporária inclui sempre o caracter # antes do nome da mesma. São usadas tabelas temporárias pois deste modo não é necessário ter uma tabela permanente e deste modo são mais eficientes em termos de espaço e performance. Nesta fase irá ocorrer uma transformação dos dados para estes não irem com incoerências para a tabela destino. Por fim passamos à fase de *loading* em que iremos depositar os dados já transformados na tabela final/destino (*dim\_candidate*). E é nesta tabela que se irá buscar os dados necessários para alimentar os indicadores. A criação das tabelas e *procedures* irá ser vista mais afundo posteriormente.

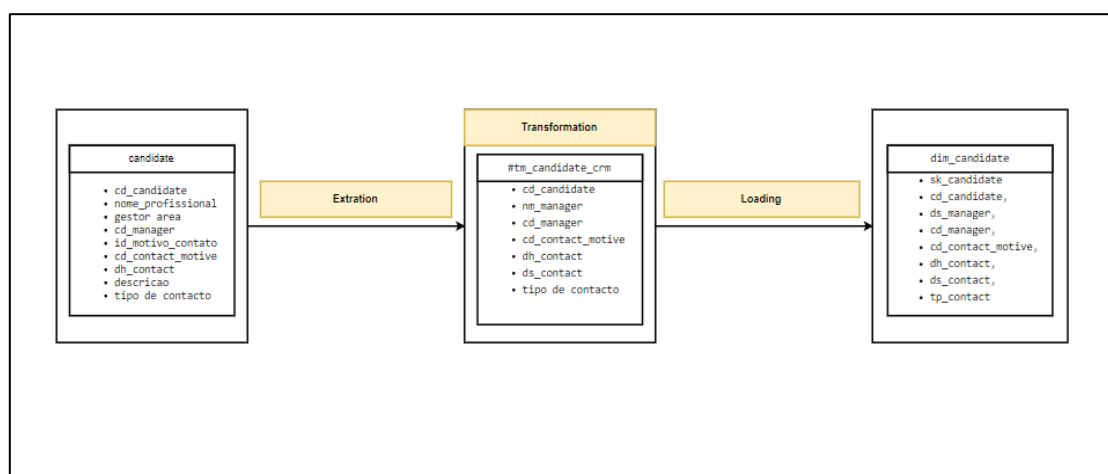


Figura 5 - Esquema ETL

A próxima paragem será a criação das páginas do *dashboard*, ou seja, dos indicadores. Para isso foi utilizada a ferramenta Power BI. O conjunto de indicadores desenvolvidos nesta ferramenta e que vão ser explicados posteriormente.

E por fim temos a última secção que é a apresentação dos indicadores aos managers. Isto tal como o levantamento de requisitos consiste em uma reunião com os managers para explicar os indicadores desenvolvidos e perceber se foram ao encontro das suas expectativas e se, se encontram satisfeitos com o trabalho desenvolvido.

Agora que já foi apresentada a organização das atividades podemos voltar para a parte mais técnica que é a criação das tabelas e *procedures*, mas desta vez de uma forma mais aprofundada.

Começando pelas tabelas podemos observar na figura 6 o script para a criação da tabela já apresentada previamente (dim\_candidate). Esta tabela inclui uma surrogate key (sk\_candidate) que é uma chave primária da tabela e é gerada automaticamente sempre que existe um registo novo na tabela graças à criação de um índice, figura 7. Esta coluna nunca irá ter um valor repetido por ser usado o IDENTITY(1,1). Isto vai fazer com que haja sempre um incremento de uma unidade no número. Depois temos várias colunas com o prefixo cd que significa código. Estes vão ser valores inteiros. Temos outros com o prefixo de ds que significa descrição. Estes vão possuir nomes ou descrições tal como o nome indica. E por fim temos um tp. Este tp significa tipo. Estes prefixos foram utilizados pois estavam presentes no documento de metodologia de base de dados. Com a utilização destes prefixos conseguimos ter uma base de dados com nomes coerentes.

```
CREATE TABLE [dbo].[dim_candidate](
    [sk_candidate] [int] IDENTITY(1,1) NOT NULL,
    [cd_candidate] [int] NULL,
    [cd_manager] [int] NULL,
    [cd_contact_motive] [tinyint] NULL,
    [dh_contact] [datetime] NULL,
    [ds_contact] [varchar](200) NULL,
    [ds_manager] [varchar](100) NULL,
    [tp_contact] [nvarchar](15) NULL,
    CONSTRAINT [sk_dim_candidate] PRIMARY KEY CLUSTERED
(
    [sk_candidate] ASC
)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF, IGNORE_DUP_KEY = OFF,
ALLOW_ROW_LOCKS = ON, ALLOW_PAGE_LOCKS = ON, OPTIMIZE_FOR_SEQUENTIAL_KEY = OFF) ON
[PRIMARY]
) ON [PRIMARY]
GO
```

Figura 6 – Tabela dim\_candidate

```
CREATE INDEX sk_dim_candidate
ON dim_candidate(sk_candidate);

GO
```

Figura 7 - Índice surrogate key candidate

Quanto à *procedure*, é aqui que acontece todo o ETL, figura 8. São extraídos os dados da tabela que se encontra na outra base de dados, depositados e transformados na tabela temporária e finalmente depois de tratados são enviados para a tabela final.

```
ALTER procedure [dbo].[spn_load_dim_candidate] as

begin
    set nocount on;
    declare @msg varchar(200), @count int;

    drop table if exists #tm_candidate;

    SELECT cast(id as int) cd_candidate, nome_profissional nm_manager,
    convert(int, [gestor area]) cd_manager, cast(id_motivo_contato as int)
    cd_contact_motive,
    convert(datetime, (format(data, 'yyyy-MM-dd') + ' ' +
    convert(varchar(20), hora, 108))) dh_contact, c.descricao as ds_contact, [tipo
    de contacto]
    into #tm_candidate
    FROM [candidate] cc
    left join [contato] c on c.id_motivo_contato = c.id_tp_motivo_contato
    left join [recursos] r on cc.[gestor area] = r.id_rec
    where [gestor area] > 0 and nome_profissional is not null

    select @count = count(0) from #tm_candidate;

    if @count > 0
        begin
            truncate table dim_candidate;

            insert into dim_candidate (cd_candidate, ds_manager,
            cd_manager, cd_contact_motive, dh_contact, ds_contact,
            tp_contact)
            select cd_candidate, nm_manager, cd_manager, cd_contact_motive,
            dh_contact, ds_contact, [tipo de contacto]
            from #tm_candidate

            ALTER INDEX sk_dim_candidate ON dim_candidate REBUILD PARTITION
            = ALL WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF,
            SORT_IN_TEMPDB = OFF, ONLINE = OFF, ALLOW_ROW_LOCKS = ON,
            ALLOW_PAGE_LOCKS = ON);

            ALTER INDEX sk_dim_candidate ON dim_candidate REORGANIZE WITH
            ( LOB_COMPACTION = ON );

            set @msg = 'Data load spn_load_dim_candidate_crm done
            successfully. ' + convert(varchar(10), @count) + ' rows.';
            RAISERROR (N'Success: %s', 10, 1, @msg);
        end
    else
        begin
            set @msg = 'Data load spn_load_dim_candidate failed.';
            RAISERROR (N'Error: %s', 10, 1, @msg);
        end
    set nocount off;
end
```

Figura 8 - Procedure candidate

Tanto de tabelas como de *procedures* foram apresentadas apenas uma de cada para não tornar o relatório muito extenso.

Na figura 9 podemos observar o bloco de código onde acontece a fase de ET (*Extract, Transform*). Estamos a selecionar as colunas da tabela candidate e a enviar para a tabela temporário onde vai ter as alterações necessárias. Uma dessas alterações é a transformação da data em formato 'yyyy-MM-dd'.

```
SELECT cast(id as int) cd_candidate, nome_profissional nm_manager,  
convert(int, [gestor area]) cd_manager, cast(id_motivo_contato as int)  
cd_contact_motive,  
convert(datetime, (format(data, 'yyyy-MM-dd') + ' ' +  
convert(varchar(20), hora, 108))) dh_contact, c.descricao as ds_contact, [tipo  
de contacto]  
into #tm_candidate  
FROM [candidate] cc  
left join [contato] c on c.id_motivo_contato = c.id_tp_motivo_contato  
left join [recursos] r on cc.[gestor area] = r.id_rec  
where [gestor area] > 0 and nome_profissional is not null  
  
select @count = count(0) from #tm_candidate;
```

Figura 9 - Bloco de código ET

Na figura 10 já nos encontramos na parte de *load*. Aqui estamos a inserir os dados na tabela destino de onde depois iremos retirar os dados para os indicadores.

```
begin  
    truncate table dim_candidate;  
  
    insert into dim_candidate (cd_candidate, ds_manager,  
cd_manager, cd_contact_motive, dh_contact, ds_contact,  
tp_contact)  
select cd_candidate, nm_manager, cd_manager, cd_contact_motive,  
dh_contact, ds_contact, [tipo de contacto]  
from #tm_candidate  
  
ALTER INDEX sk_dim_candidate ON dim_candidate REBUILD PARTITION  
= ALL WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF,  
SORT_IN_TEMPDB = OFF, ONLINE = OFF, ALLOW_ROW_LOCKS = ON,  
ALLOW_PAGE_LOCKS = ON);  
  
ALTER INDEX sk_dim_candidate ON dim_candidate REORGANIZE WITH  
( LOB_COMPACTION = ON );  
  
set @msg = 'Data load spn_load_dim_candidate_crm done  
successfully. ' + convert(varchar(10), @count) + ' rows.';  
RAISERROR (N'Success: %s', 10, 1, @msg);  
end
```

Figura 10 - Bloco de código L

Como mencionado anteriormente vão ser analisado em mais detalhe os indicadores e como os mesmo foram construídos.

Começando pela primeira página, figura 11 (página de contactos), podemos observar 4 indicadores: candidatos distintos, total contactos, distribuição de contactos por motivo de contacto (gráfico pizza) e o top 10 managers. Por motivos de confidencialidades os nomes dos managers não podem ser mostrados.

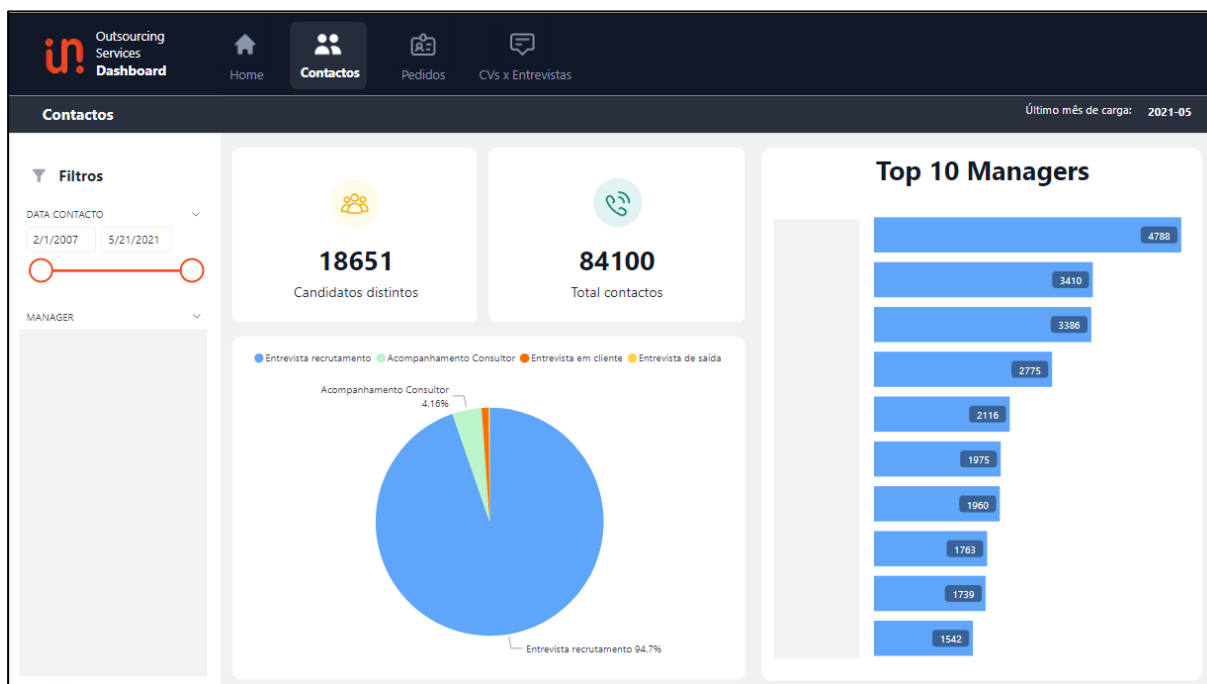


Figura 11 - Página de Contactos

Começando pelo indicador de Candidatos distintos, este indicador, como o nome indica, tem o objetivo o total de candidatos contactos num dado período de tempo. Para isto foi necessário a utilização de DAX para a construção de uma *measure*, figura 12. Foi necessário utilizar a função `DISTINCTCOUNT()` para se contabilizar apenas os distintos pois senão iríamos obter o total de contactos. Podemos observar que temos 18651 candidatos distintos foram contactados por managers.

```
Candidates_distinct = DISTINCTCOUNT(dim_candidate[cd_candidate])
```

Figura 12 - Candidatos distintos



Para o indicador Total contactos é muito similar ao anterior. Só que em vez de se utilizar a função `DISTINCTCOUNT()` utiliza-se a função `COUNT()`, figura 13. Podemos observar que os managers fizeram um total de 84100 contactos.

```
Total_Contacts = COUNT(dim_candidate[cd_candidate])
```

Figura 13 - Total contactos

Avançando para o gráfico pizza, distribuição de contactos por motivo de contacto este foi bastante básico pois foi só seleccionar o tipo de gráfico e introduzir o campo que queríamos. E têm como objetivo mostrar a proporção de cada motivo de contacto, sendo que o predominante é “Entrevistas de recrutamento”. Mas este indicador também inclui um tooltip (pop up que abre quando se passa o cursor sobre um elemento da interface), figura 14. Aqui podemos observar o total de contactos por tipo de contacto, ou seja, se foi telefónico, e-mail, automático, etc. E percebemos que o tipo predominante é e-mail.

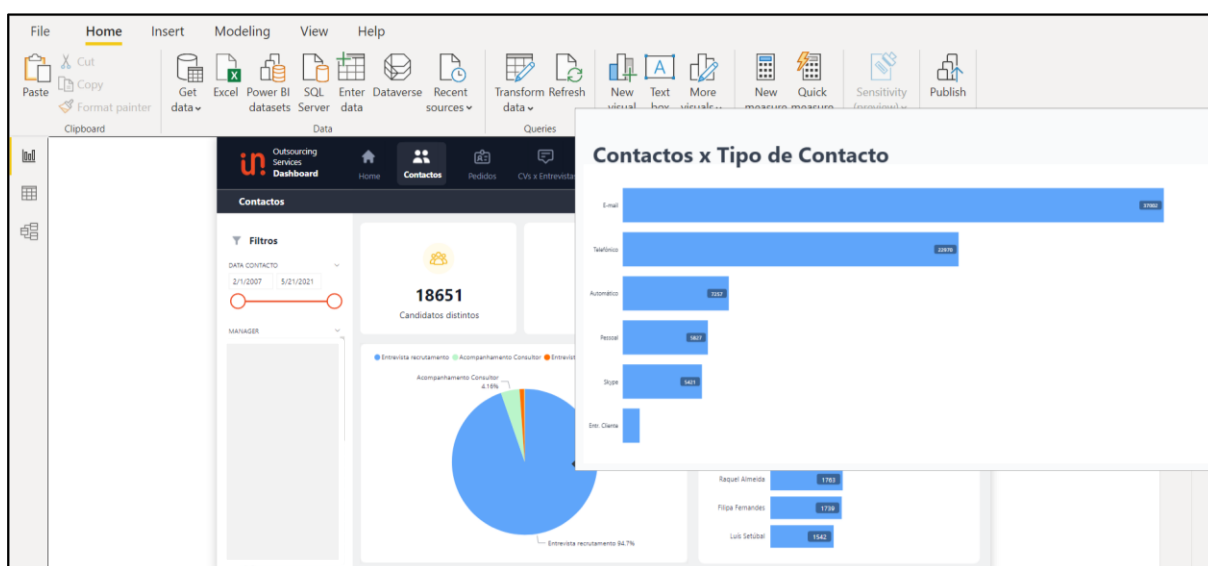


Figura 14- Tooltip Contactos x Tipo de Contacto

Por fim quanto a esta página apenas falta falar de um indicador, top 10 managers em termos de contactar candidatos distintos. A *measure* utilizada encontra-se na figura 15 e foi utilizada a função `RANKX()` para construir o ranking. Neste indicador por questões de privacidade não podem ser mostrados o nome dos managers. Porém

podemos observar que o que se encontra em primeiro lugar contactou 37002 candidatos distintos.

```
top10_managers_cand =
VAR ManagerRank = RANKX(ALL(dim_candidate[ds_manager]),
CALCULATE(DISTINCTCOUNT(dim_candidate[cd_candidate])),,DESC)
Return
IF(ManagerRank <= 10, CALCULATE(DISTINCTCOUNT(dim_candidate[cd_candidate])), BLANK())
```

Figura 15 - DAX Contactos x Tipo de Contacto

Passando para a página de Pedidos, figura 16, nesta página foram solicitadas 4 métricas. Sendo estas total de pedidos, total de currículos, total de respostas e total de respostas x total de currículos por manager. Como mencionado previamente, por motivos de confidencialidade, os nomes dos managers não podem estar visíveis.

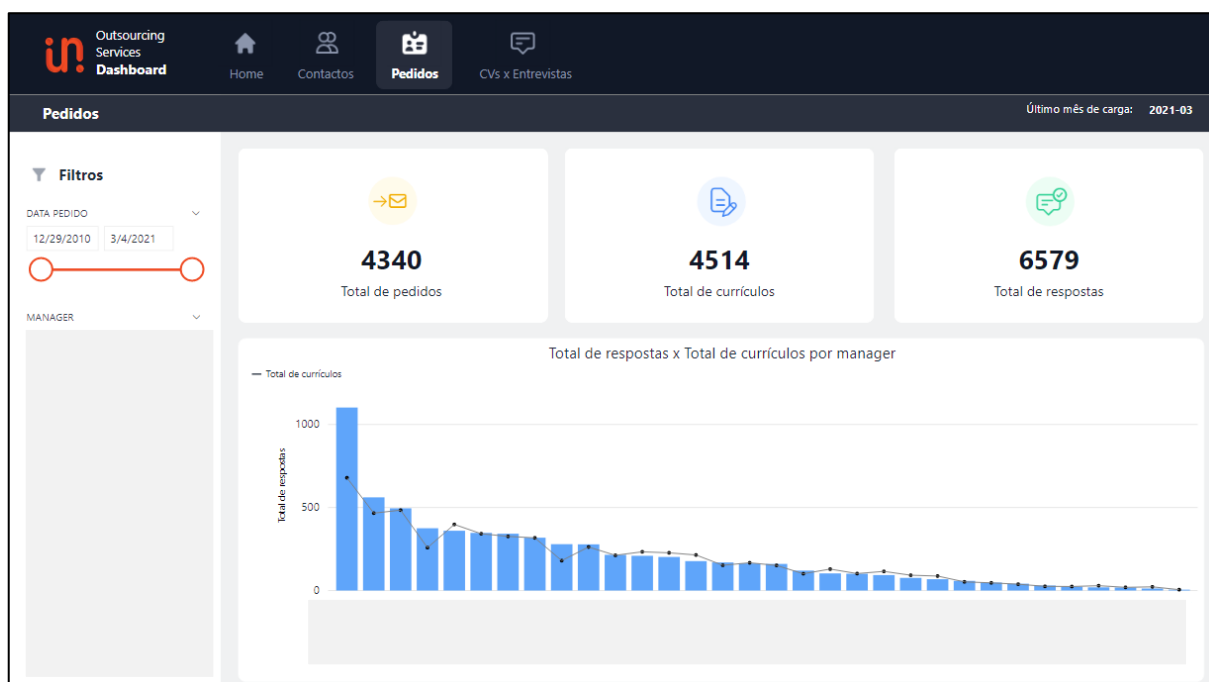


Figura 16 - Página de Pedidos

Para a construção destas métricas foram necessários dados de 3 tabelas, dim\_request, dim\_proposal e dim\_proposal\_candidate. Todas estas tabelas seguem a metodologia de base de dados mencionada anteriormente.

Na primeira métrica, Total de pedidos, os dados foram extraídos da dim\_request. Para a construção desta métrica foi apenas necessário arrastar o campo que pretendíamos (cd\_request) e selecionar o Count(Distinct), figura 17, para contabilizar os cd\_request distintos, ou seja, o total de pedidos. São pedidos de candidatos para participarem em vagas.

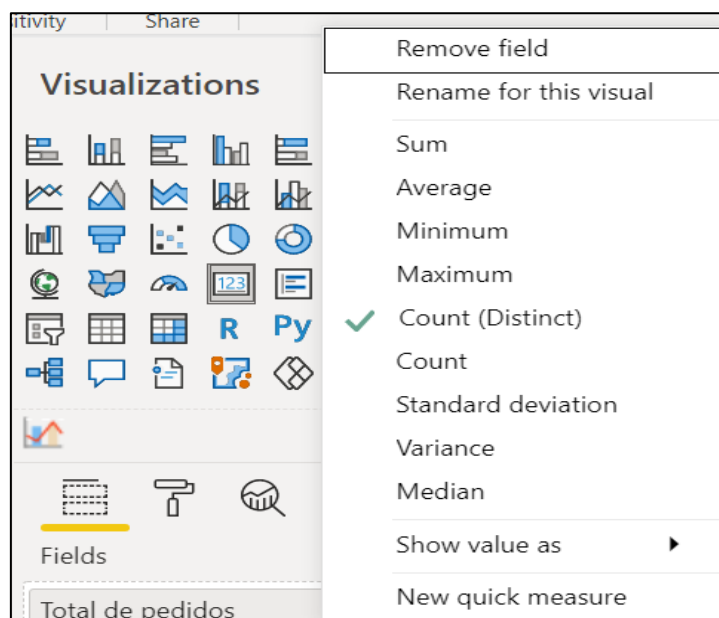


Figura 17 - Total de pedidos

Na segunda métrica, Total de currículos, os dados foram extraídos da dim\_proposal\_candidate. Para a construção desta métrica foi o mesmo que na métrica anterior, ou seja, arrastar o campo que pretendíamos (cd\_proposal\_cand) e selecionar o Count(Distinct). O total de currículo é o total de currículos enviados por managers a clientes.

Avançamos agora para a terceira métrica, Total de respostas. Nesta métrica mais uma vez o procedimento foi o mesmo que nas métricas anteriores. Porém para a construção desta métrica foi necessário a extração dos dados a partir da tabela dim\_proposal. E utilizámos a coluna cd\_proposal. O total de respostas contabiliza o total de respostas recebidas de clientes.

Para a quarta e última métrica desta página temos o gráfico Total de respostas x Total de currículos por manager. Nesta métrica analisar a proporção de respostas

para o número de currículos enviados. Esta métrica possui ainda um tooltip, figura 18, que permite visualizar o top 10 clientes em termos de pedidos.

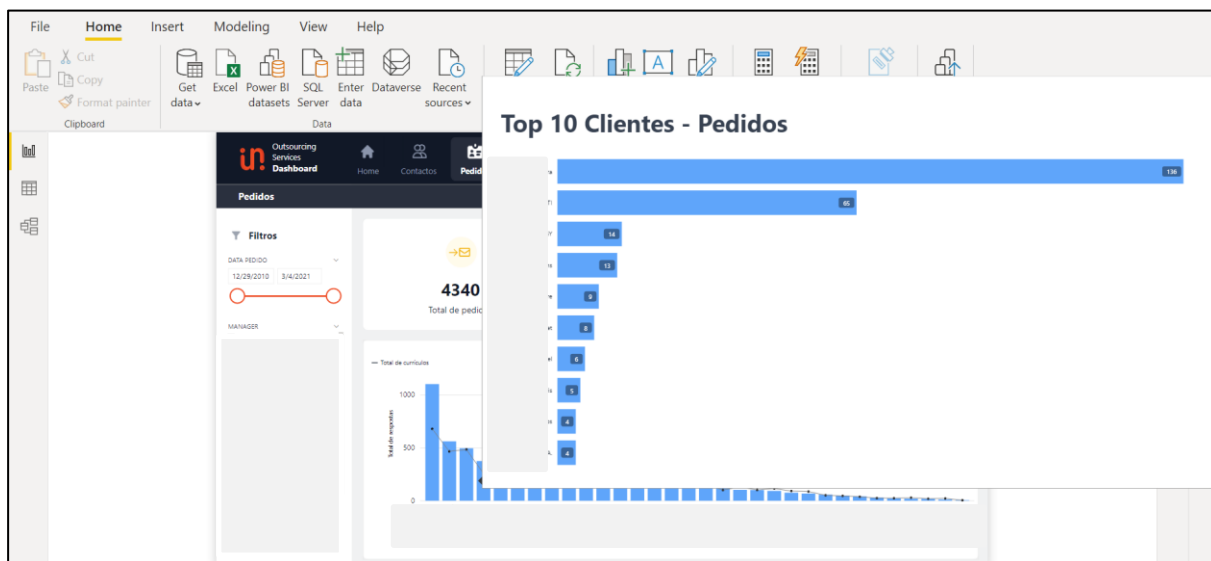


Figura 18 - Tooltip top 10 clientes (pedidos)

Por último a última página desenvolvida durante o estágio, a página de CVs x Entrevistas (figura 19). Nesta página foram solicitadas 4 métricas. O total de entrevistados, rejeitados e adjudicados e o rácio entre entrevistados e adjudicados. As outras 3 métricas foram ideias sugeridas pela equipa que foram muito bem aceites.

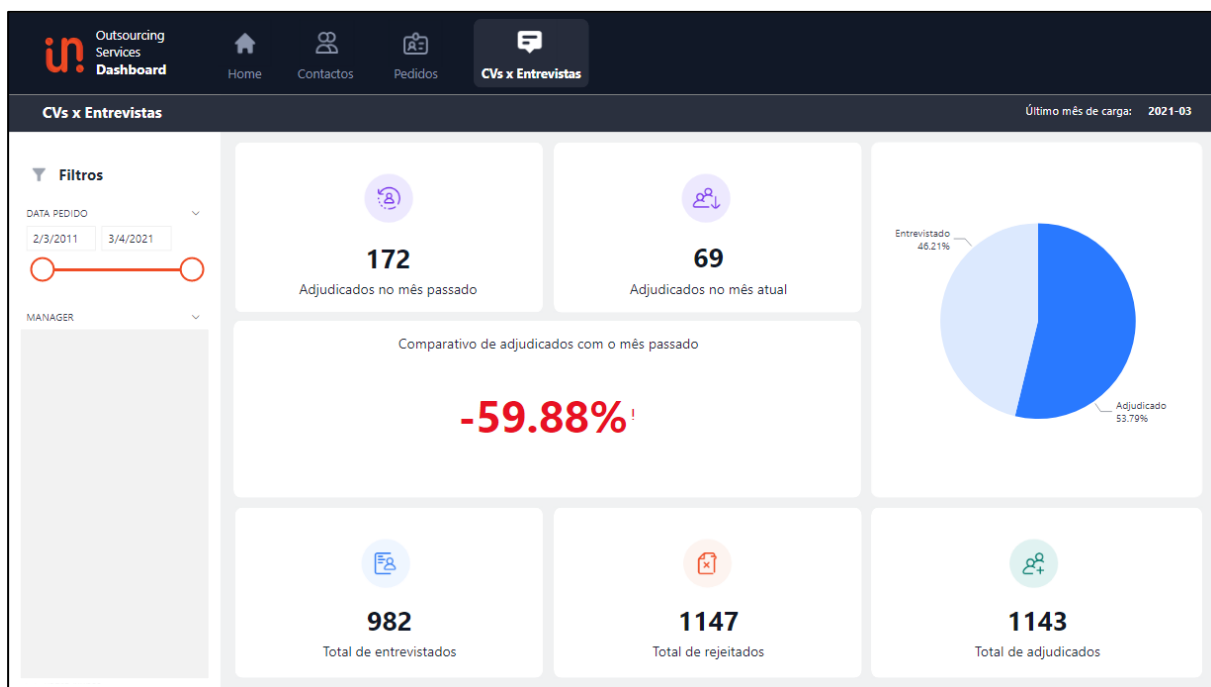


Figura 19 - Página CVs x Entrevistas

Começando pelo total de entrevistados para isto foi necessário a criação de uma *measure*, figura 20. Nesta *measure* vai haver a contabilização de todos os *cd\_request* que são distintos e caso o valor seja 0, em vez de devolver BLANK, irá devolver 0. Para a métrica apenas mostrar os respectivos entrevistados foi necessário selecionar na secção de *filters* que apenas pretendíamos o *ds\_status(groups) = Entrevistado*, figura 21.

```
count_id =
VAR count_id = CALCULATE(DISTINCTCOUNT(dim_request[cd_request]))
Return
IF(count_id = 0, 0, CALCULATE(DISTINCTCOUNT(dim_request[cd_request])))
```

Figura 20 - *measure* totais

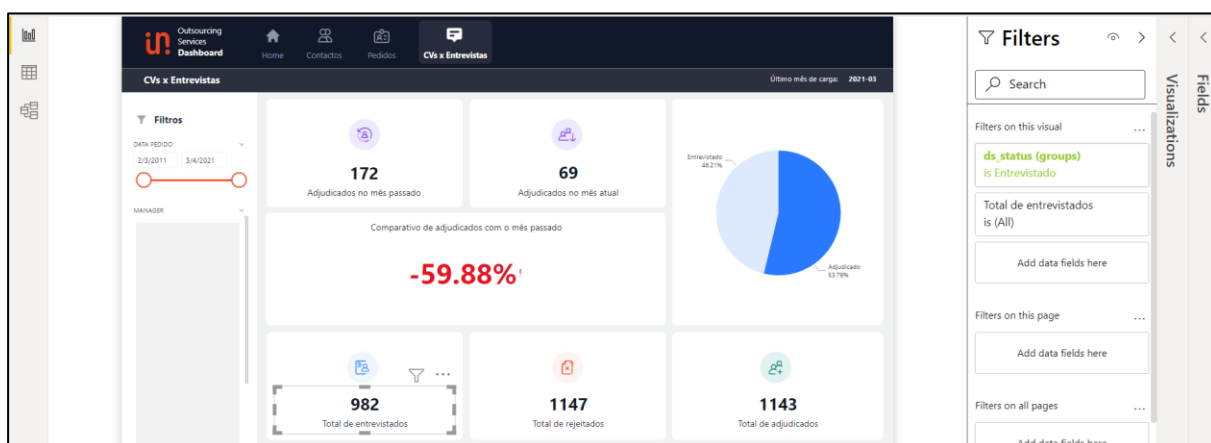


Figura 21 - Selecionar filtro

Quanto as outras duas métricas de totais, total de rejeitados e total de adjudicados o procedimento é o mesmo que para a métrica anterior. Foi utilizado a *measure* da figura 21 e selecionado o respetivo filtro, ou seja, para o total de rejeitados foi necessário selecionar *ds\_status(groups) = rejeitado* e para total de adjudicados foi necessário selecionar *ds\_status(groups) = adjudicado*.

Quanto à métrica para ver o rácio entre entrevistado e adjudicado foi apenas necessário selecionar o tipo de gráfico, figura 22, e selecionar os grupos que pretendíamos ter no gráfico, entrevistado e adjudicado, figura 23.

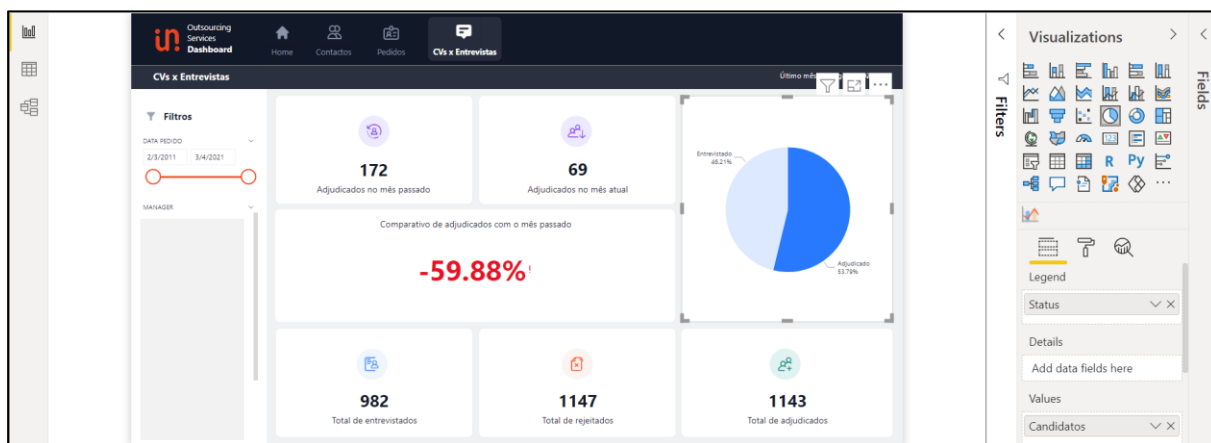


Figura 22 - Selecionar tipo de visualização

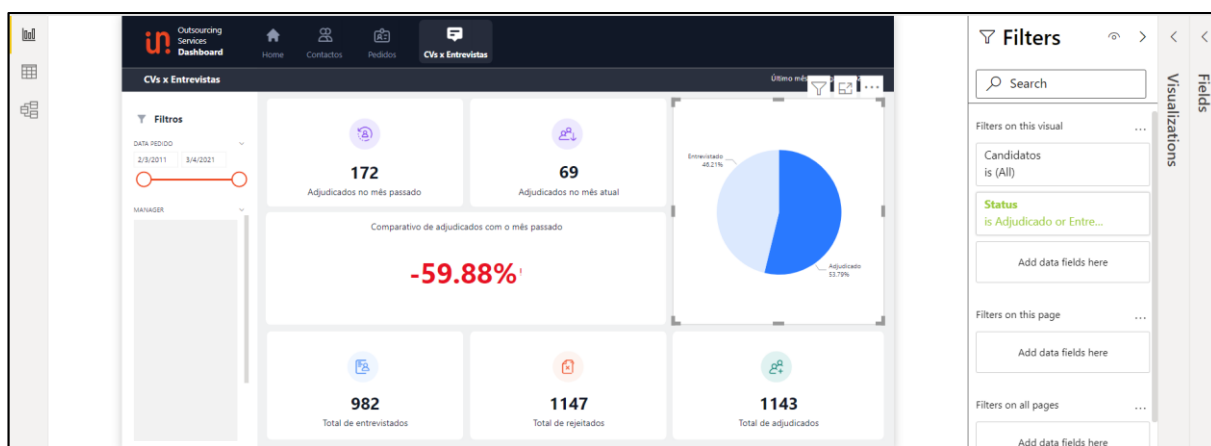


Figura 23 - Selecionar filtro para o gráfico

Avançamos agora para as métricas que foram propostas pela equipa, adjudicados no mês passado, adjudicados no mês atual e comparativo de adjudicados com o mês passado.

Começando pela métrica de adjudicado no mês passado. Nesta métrica foi necessário uma *measure*, figura 24. Esta *measure* irá dar-nos o total de adjudicados do mês anterior pois na variável "PreviousDate" definimos o mês anterior retirando uma unidade à CurrentDate é que o último registo da base de dados. Podemos observar que tivemos 172 adjudicados no mês anterior.

```

prev_month_adju =
VAR CurrentDate = MAX(fact_billing[date_reference])
VAR PreviousDate = DATE(Year(CurrentDate),Month(CurrentDate)-1, Day(CurrentDate))
VAR PrevMonthAdju =
CALCULATE(DISTINCTCOUNT(fact_billing[cd_ic]),fact_billing[date_reference] =
PreviousDate)
RETURN
PrevMonthAdju

```

*Figura 24 - Measure adjudicados no mês passado*

Na métrica seguinte, “adjudicados no atual”, foi necessário mais uma *measure*, figura 25. Nesta apenas utilizamos a variável CurrentDate pois esta vai dar-nos os registos do mês atual. Podemos observar que tivemos 69 adjudicados no mês atual.

```

adju_current =
VAR CurrentDate = MAX(fact_billing[date_reference])
VAR ActualMonthAdju = CALCULATE(DISTINCTCOUNT(fact_billing[cd_ic]),
fact_billing[date_reference] = CurrentDate)
RETURN
ActualMonthAdju

```

*Figura 25 - Measure adjudicado no mês atual*

E por último, comparativo de adjudicados com mês passado. Com esta métrica podemos observar a evolução do número de adjudicados no mês atual comparando ao mês anterior. Para a construção desta métrica foi necessário a criação de uma *measure*, figura 26. Isto vai nos dar a percentagem de comparativamente ao mês anterior. Depois foi apenas selecionar o gráfico e os campos necessários. Assim podemos observar que houve um decréscimo de quase 60% em comparação ao mês passado, figura 27.

```

adju_growth = ([adju_current]/[prev_month_adju] - 1)

```

*Figura 26 - Measure comparativo com adjudicados mês passado*

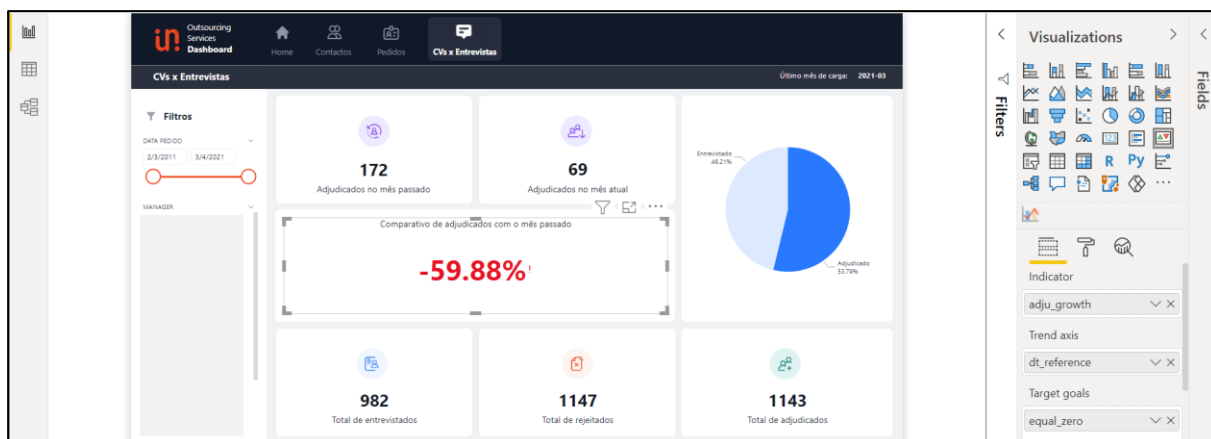


Figura 27 - Seleção do gráfico e campos necessários

A Integer Consulting sempre teve como preocupação a melhoria contínua. Face à análise na gestão do ciclo de entrevista dos candidatos e na gestão de pedidos de profissionais pelos seus clientes, estabeleceu a necessidade de criação de um conjunto de indicadores. Estes indicadores vêm contribuir para facilitar o processo de tomada de decisão por parte dos gestores da Integer. No total. Ao longo do estágio, foram desenvolvidas três páginas no *dashboard*, três *procedures* e três tabelas.

Todos os indicadores criados cumprem os requisitos/necessidades da empresa e estes indicadores representam um papel fundamental no processo de tomada de decisão. Estes indicadores foram possíveis desenvolver durante o período de estágio e aprovados pela empresa. Este projeto sofreu bastantes alterações especialmente em termos de design em que houve a intervenção de um designer para a construção de uma *mockup* para se poder ter uma interface mais *user friendly*. Com esta intervenção foi possível entregar um produto com melhor qualidade e que vai ao encontro dos padrões da empresa.



## 7 Conclusões

A construção dum sistema de indicadores abriu a oportunidade para o desenvolvimento de um projeto de BI, quase do zero, e ajudar os gestores da Integer Consulting na tomada de decisão. Foi dada a oportunidade de fazer o levantamento de requisitos, criação de documentação acerca dos indicadores e a criação dos respetivos indicadores.

Inicialmente surgiram algumas dificuldades, nomeadamente na parte da criação das *stored procedures*, pois nunca tinha trabalhado com essa funcionalidade. Mas com a ajuda de colegas de trabalho, estas dificuldade foram ultrapassadas muito rapidamente. Na utilização do Power BI não houve muitos obstáculos ao longo do caminho.

As unidades curriculares frequentadas na licenciatura de Bioinformática contribuíram para o sucesso neste estágio curricular. Contribuíram tanto com conhecimento teórico como prática. Apesar de ter sido necessária a aprendizagem de *stored procedures*, o resto das tecnologias e técnicas abordadas durante a licenciatura foram bastante benéficas. Os projetos desenvolvidos nas unidades curriculares também contribuíram para ideias implementadas.

A realização deste estágio permitiu enriquecer o conhecimento em BI, bem como aprofundar conhecimentos em termos de realização de consultas em SQL. Contribuiu para o desenvolvimento de uma atitude mais madura e profissional. Para além disso contribuiu para uma atitude proativa, sempre à procura de formas de melhorar.

Em suma, foi possível adquirir imensa experiência que contribuiu para a aprendizagem de novas funcionalidade e técnicas que são utilizadas todos os dias no mundo do trabalho.

## 8 Conclusions

The construction of a system of indicators opened the opportunity for the development of a BI project, almost from scratch, and to help Integer Consulting managers in the decision-making process. The opportunity was given to gather requirements, create documentation about the indicators and create the respective indicators.

Initially, some difficulties arose, namely in the creation of stored procedures, as I had never worked with this functionality. But with the help of co-workers, these difficulties were overcome very quickly. In using Power BI there were not many obstacles along the way.

The curricular units attended in the Bioinformatics degree contributed to the success of this curricular internship. They contributed both theoretical and practical knowledge. Although it was necessary to learn stored procedures, the rest of the technologies and techniques covered during the degree were very beneficial. The projects developed in the curricular units also contributed to some of the implemented ideas.

The completion of this internship allowed enriching the knowledge in BI, as well as deepening knowledge in terms of performing SQL queries. Contributed to the development of a more mature and professional attitude. Furthermore, it contributed to a proactive attitude, always looking for ways to improve.

In short, it was possible to acquire immense experience that contributed to the learning of new features and techniques that are used every day in the world of work.

## Referências

- [1] “Jasmin Software,” Mariana Gomes, 12 3 2020. [Online]. Available: <https://www.jasminsoftware.pt/blog/business-intelligence/>. [Acedido em 03 07 2021].
- [2] S. S. Bhimasankaram Pochiraju, Essentials of Business Analytics An Introduction to the Methodology and its Applications, 2019.
- [3] “SAS,” [Online]. Available: [https://www.sas.com/pt\\_br/insights/data-management/o-que-e-etl.html](https://www.sas.com/pt_br/insights/data-management/o-que-e-etl.html). [Acedido em 03 07 2021].
- [4] M. R. Ralph Kimball, The Data Warehouse Toolkit, Wiley, 2013.
- [5] “Dimension Table Structure,” [Online]. Available: <https://www.kimballgroup.com/data-warehouse-business-intelligence-resources/kimball-techniques/dimensional-modeling-techniques/dimension-table-structure/>. [Acedido em 12 07 2021].
- [6] “Fact Table Structure,” [Online]. Available: <https://www.kimballgroup.com/data-warehouse-business-intelligence-resources/kimball-techniques/dimensional-modeling-techniques/fact-table-structure/>. [Acedido em 12 07 2021].
- [7] “Accumulating Snapshot Fact Table,” [Online]. Available: <https://www.kimballgroup.com/data-warehouse-business-intelligence-resources/kimball-techniques/dimensional-modeling-techniques/accumulating-snapshot-fact-table/>. [Acedido em 12 07 2021].
- [8] “Periodic Snapshot Fact Table,” [Online]. Available: <https://www.kimballgroup.com/data-warehouse-business-intelligence-resources/kimball-techniques/dimensional-modeling-techniques/periodic-snapshot-fact-table/>. [Acedido em 12 07 2021].
- [9] “Surrogate Key vs Natural Key Differences and When to Use in SQL Server,” [Online]. Available: <https://www.mssqltips.com/sqlservertip/5431/surrogate-key-vs-natural-key-differences-and-when-to-use-in-sql-server/>. [Acedido em 13 07 2021].
- [10] “7 reasons to use Microsoft Power BI,” [Online]. Available: <https://www.stitchdata.com/resources/7-reasons-power-bi/>. [Acedido em 2021 07 14].

- [11] "SQL Server Management Studio (SSMS)," [Online]. Available: <https://docs.microsoft.com/en-us/sql/ssms/sql-server-management-studio-ssms?view=sql-server-ver15>. [Acedido em 14 07 2021].
- [12] "What Is Scrum Methodology? & Scrum Project Management," [Online]. Available: <https://www.digite.com/agile/scrum-methodology/>. [Acedido em 14 07 2021].
- [13] "User Stories," [Online]. Available: <https://www.atlassian.com/agile/project-management/user-stories>. [Acedido em 14 07 2021].
- [14] F. M. Fowler, Navigating Hybrid Scrum Environments: Understanding the Essentials, Avoiding the Pitfalls, 2018.
- [15] "Scrum.org," [Online]. Available: <https://www.scrum.org/resources/what-is-sprint-planning>. [Acedido em 20 07 2021].
- [16] S. Ockerman, "Scrum.org," [Online]. Available: <https://www.scrum.org/resources/what-is-a-daily-scrum>. [Acedido em 20 07 2021].
- [17] "Scrum.org," [Online]. Available: <https://www.scrum.org/resources/what-is-a-sprint-review>. [Acedido em 20 07 2021].
- [18] "Scrum.org," [Online]. Available: <https://www.scrum.org/resources/what-is-sprint-planning>. [Acedido em 20 07 2021].
- [19] A. Abreu, "Metodologia de Base de Dados".
- [20] A. Abreu, "Metodologia de BI".
- [21] "Kimball Group," [Online]. Available: <https://www.kimballgroup.com/data-warehouse-business-intelligence-resources/kimball-techniques/dimensional-modeling-techniques/dimension-surrogate-key/>. [Acedido em 12 07 2021].