FREE-OSINT

Uma ferramenta modular para aquisição e gestão dos dados de fontes abertas

Mestrado em Cibersegurança e Informação Forense

Vladyslav Adamovych

Leiria, novembro 2021

FREE-OSINT

Uma ferramenta modular para aquisição e gestão dos dados de fontes abertas

Mestrado em Cibersegurança e Informação Forense

Vladyslav Adamovych

Trabalho de Projeto realizado sob a orientação do Professor Doutor Leonel Santos.

Leiria, novembro 2021

Originalidade e Direitos de Autor

O presente relatório de projeto é original, elaborado unicamente para este fim, tendo sido devidamente citados todos os autores cujos estudos e publicações contribuíram para o elaborar.

Reproduções parciais deste documento serão autorizadas na condição de que seja mencionado o Autor e feita referência ao ciclo de estudos no âmbito do qual o mesmo foi realizado, a saber, Curso de Mestrado em Cibersegurança e Informação Forense, no ano letivo 2020/2021, da Escola Superior de Tecnologia e Gestão do Instituto Politécnico de Leiria, Portugal, e, bem assim, à data das provas públicas que visaram a avaliação destes trabalhos.

Agradecimentos

A realização do seguinte relatório e projeto foi possível com a ajuda de diversos indivíduos aos quais sinto a necessidade de expressar o meu agradecimento.

Ao Professor Doutor Leonel Santos pela excelente orientação, disponibilidade e apoio no trabalho desenvolvido.

Ao Professor Baltazar Rodrigues pela disponibilidade e sugestões oferecidos durante o processo de desenvolvimento.

Os restantes professores do mestrado pela excelência no ensino e resiliência demonstrados durante a pandemia.

À Instituição pela oportunidade na formação académica e a possibilidade participação no projeto.

Por fim quero expressar o meu agradecimento a minha família e especialmente a minha namorada Carina pelo apoio emocional e ajuda na revisão do documento.

Muito obrigado.

Resumo

O progresso tecnológico das últimas décadas tem vindo a reforçar os efeitos da globalização, provocando reduções de custos em várias tecnologias. O público ganhou acesso a meios de comunicação que atualmente são considerados indispensáveis. Estes meios permitiram partilhar todo o tipo de informação e mudar a interação social. A quantidade de informação que se tornou pública expandiu as capacidades da comunidade de inteligência, mas trouxe novos desafios.

Neste documento é estudado o mundo digital de OSINT. Para desenvolver uma perspetiva correta acerca do estado atual da OSINT digital são analisadas as fontes, ferramentas, ambientes e técnicas atuais mais populares. A análise realizada revelou os aspetos positivos e negativos das soluções disponíveis, o que levou a reflecção acerca das atuais necessidades da comunidade de inteligência.

O ambiente web está em constante mudança, surgem novos serviços e fontes de OSINT, alguns perdem relevância no mercado, mas mantêm viabilidade para os investigadores. Para além disto, estão regularmente a surgir novas técnicas e ferramentas para adquirir inteligência tanto para as novas como as atuais fontes abertas. A maioria das ferramentas de OSINT gratuitas não possui caraterísticas de uma ferramenta intuitiva, nem oferece opções para a gestão da informação, apenas se focando na extração de um formato específico da inteligência com interfaces em linhas de comando. A perspetiva que surge disto é a descentralização e inconsistência nas soluções de OSINT onde, o investigador necessita de uma ferramenta para cada serviço e é responsável pela agregação da informação num ambiente externo para poder criar alguma coerência e ligação nos dados.

Com esta perspetiva em mente, foi concebido um conjunto de objetivos que uma ferramenta focada no processo de OSINT deverá seguir. Estes objetivos foram disseminados em requisitos e caraterísticas chave que, de seguida foram aplicados no desenvolvimento da estrutura de uma solução proposta. Para além disso, a solução foi desenvolvida tendo em conta as limitações e os aspetos positivos de ferramentas atuais, o que resultou em caraterísticas principais como a arquitetura modular, interação intuitiva e a capacidade de gestão da informação.

A validação da solução foi feita através da interpretação dos mecanismos desenvolvidos face aos objetivos definidos, bem como, distribuições da solução para um conjunto de pessoas próximas da área de OSINT. A apresentação dos mecanismos desenvolvidos em conjunto com o *feedback* positivo dos entrevistados permitiu concluir que os objetivos definidos foram concretizados, a solução proposta contribuirá para a comunidade de inteligência e cibersegurança, e espera-se que esta poderá guiar o desenvolvimento de OSINT numa direção mais focada.

Palavras-chave: OSINT, Inteligência, Open Source

Abstract

The technological progress of the last few decades has reinforced the effects of globalization by allowing access to different technologies and lowering their cost. The public has gained access to communication technologies, considered essential by today's standards. These technologies allow their users to share information, thus changing social interaction. The amount of information that has become available to the public expanded intelligence community's capabilities but also brought new challenges.

This document studies the digital world of OSINT. To develop a correct perspective on the current state of digital OSINT we study the most popular sources, tools environments and techniques. The analysis revealed positive and negative aspects of currently available solutions, which made us reflect on the ongoing necessities of the intelligence community.

The web environment is at constant change, new services and sources for OSINT emerge, some lose their relevance in the market, but nevertheless continue as being relevant for investigators. In addition to the forementioned, new techniques and tools originate regularly, that acquire intelligence from the new and current sources. Most free OSINT tools don't possess characteristics of an intuitive application, nor do they offer data management options, focusing only on specific data extraction using command line interfaces. This creates a perspective of decentralization and inconsistency of OSINT solutions where an investigator requires a tool for each service and is responsible for aggregating the information on external environments to create some coherence and link data.

With this perspective in mind, we generated a set of objectives that an OSINT focused tool must follow. These objectives were then disseminated into key requirements and characteristics, that were applied in the development of the structure of a proposed solution. Besides, the solution was developed accounting for the limitations and positive aspects of modern tools, which resulted in characteristics like modular architecture, intuitive interaction and data management capability.

Validation for the solution was done through interpreting implemented mechanisms in relation to defined objectives, as well as distributions of the tool to a set of people close to the OSINT area. The analysis of the developed mechanisms in addition to the positive feedback from the reviewed party lead to a conclusion that the current solution meets the

previously defined goals, will contribute to the intelligence community and hopefully guide the development into a more focused direction.

Keywords: OSINT, Intelligence, Open Source

Índice

Lista de Figuras

Lista de siglas e acrónimos

API	Application Programming Interface
ASN	Autonomous System Number
BSD	Berkeley Source Distribution
CEF	Cromium Embedded Framework
COMINT	Communications Intelligence
DFIR	Digital Forensics & Incident Response
ESTG	Escola Superior de Tecnologia e Gestão
GDPR	General Data Protection Regulation
GUI	Graphical User Interface
HDPI	High Density Pixel Image
HTML	HyperText Markup Language
HUMINT	Human Intelligence
IMINT	Imagery Intelligence
IP	Internet Protocol
IPLeiria	Instituto Politécnico de Leiria
ISP	Internet Service Provider
JSON	JavaScript Object Notation
LTS	Long Term Support
MASINT	Measurements and Signature Intelligence
MCIF	Mestrado de Cibersegurança e Informação Forense
OSINT	Open Source Intelligence
PAT	Personal access token
PPC	Pay-Per-Click
SEO	Search Engine Optimization
SERP	Search Engine Result Page
SIGNIT	Signal Intelligence
TOR	The Onion Router
URL	Uniform Resource Locator
WWW	World Wide Web
XML	Extensible Markup Language

1. Introdução

Segundo o U.S. Naval War College, a inteligência adquirida pode ser classificada como: a coleção de informação obtida através de recursos humanos, designada por *Human Intelligence,* ou HUMINT, transmissões eletrónicas recolhidas pelas naves, aviões, satélites e outros, designados por *Signals Intelligence,* ou SIGINT, informação em forma de imagens ou fotografias, designada por *Imagery Intelligence,* ou IMINT/PHOTINT, representação visual de atividades terrestres, designada por *Geospacial Intelligence,* ou GEOINT, informação acerca de capacidades dos armamentos e atividades industriais, designada por *Measurement and Signatures Intelligence,* ou MASINT e por fim, a informação que abrange as fontes abertas em vários formatos disponíveis ao público, designada por *Open Source Intelligence*, ou OSINT [1].

O general do exército dos Estados Unidos da America, W.F. Kernan, afirma que "a informação obtida através de fontes públicas serve de fundamento para as outras disciplinas de inteligência e a sua aplicação sistemática pode reduzir a procura por informação classificada, limitando esta a questões que não podem ser respondidas através de fontes públicas" [2].

Para a área de cibersegurança o OSINT não é menos importante pois, é diretamente relacionado com técnicas de *footprinting*, ou reconhecimento. A aplicação de técnicas de aquisição de OSINT em contexto de cibersegurança permite identificar vulnerabilidades em forma de informações que são desnecessariamente partilhadas para o público [3]. Estas informações então, podem ser utilizadas para definir uma estratégia de penetração dos sistemas de segurança e são muitas vezes utilizadas pelas *red teams,* equipas que se dedicam à avaliação da robustez dos mecanismos de segurança através de testes de penetração.

1.1. Motivação e objetivos

A ansiedade acerca da partilha dos dados privados nos dias de hoje não é exclusiva apenas às entidades governamentais e organizações, mas também à população mundial. Os recentes eventos escandalosos como: a partilha dos dados dos utilizadores entre o Facebook e a Cambridge Analytica [4]; as acusações dos dispositivos da Huawei na possibilidade de conter *backdoors* [5]; ou a falta de robustez nos mecanismos de segurança da Zoom [6];

despertaram o interesse da população acerca da sua privacidade e dos dados que estão disponíveis ao público. Vivemos na era da informação onde, os utilizadores dum serviço web e os seus dados são vistos como um produto [7], e devido à evolução rápida das tecnologias e mecanismos de comunicação, torna-se difícil desenvolver a sensibilização necessária para proteger os dados pessoais. É mais importante que nunca reconhecer a informação existente em fontes abertas para tomar medidas na prevenção das partilhas desnecessárias.

Por este motivo, será explorado o mundo digital de OSINT, e tentar-se-á desenvolver um contributo para as comunidades de inteligência e cibersegurança, sem excluir ao mesmo tempo o utilizador comum, isto para permitir integrá-lo no mundo de OSINT e oferecer a possibilidade de melhoria da segurança da sua informação na Internet.

Consoante as motivações descritas, o projeto irá aplicar os seguintes objetivos:

- Investigar as questões relacionadas com o processo de aquisição de OSINT no mundo digital, identificar os aspetos fortes e as dificuldades associadas.

Para que seja possível contribuir á comunidade de inteligência, é necessário identificar os obstáculos numa investigação de OSINT moderna. Isto permitirá desenvolver uma consciência adequada para refletir acerca dos elementos do processo de aquisição que necessitem de otimização ou reconsideração.

- Desenvolver uma estrutura ou arquitetura de solução que irá responder às dificuldades identificadas.

O conhecimento obtido ao cumprir o objetivo anterior resultará em sugestões ou ideias concretas, que poderão tomar a forma de mecanismos ou estruturas em ferramentas e solucionar as dificuldades encontradas.

- Implementar uma solução com a arquitetura proposta e validá-la em relação às dificuldades identificadas.

Após definição de uma arquitetura que responderá às dificuldades encontradas, será necessário validá-la através da implementação da solução e confronto desta com os problemas lhe relacionados.

1.2.Metodologia

O desenvolvimento do projeto dividir-se-á em duas fases distintas. A primeira fase caracteriza-se pela pesquisa exploratória através do levantamento de material científico associado, extração dos termos principais, história, definição do processo e dos objetivos de OSINT. De seguida, serão estudados os meios e fontes de obtenção de OSINT mais populares. Para estes, será realizada uma análise comparativa, que permitirá compreender as especificidades de cada e o seu respetivo lugar no mercado. Para além do mencionado, a análise comparativa das fontes, ferramentas e ambientes permitirá identificar dificuldades no processo de aquisição e definir um conjunto de requisitos e caraterísticas que a solução a desenvolver deverá cumprir. Tendo estes definidos, iniciar-se-á a fase de desenvolvimento da solução proposta. O início desta seguirá a metodologia de cascata com os requisitos e características iniciais como prioridade. Quando os requisitos da aplicação se consideram cumpridos, será realizada a validação, que se fará através da partilha da solução e posteriores entrevistas, dando início ao processo de desenvolvimento ágil, através de incrementos com o *feedback* recolhido.

1.3.Estrutura do documento

No capítulo 2, de acordo com a metodologia, encontra-se o estudo exploratório do tópico *open-source intelligence* na era digital.

A definição dos requisitos, metodologias de desenvolvimento e a implementação da solução proposta encontram-se no capítulo 3, FREE-OSINT.

A capacidade de a aplicação cumprir os requisitos lhe atribuídos é estudada e apresentada no capítulo 4.

Por fim, no capítulo 5, serão tiradas as conclusões do trabalho realizado, em conjunto com as principais contribuições, limitações e tópicos para o trabalho futuro.

2. Open-source Intelligence na era digital

Neste capítulo serão estudados vários tópicos, começando com a história de OSINT, os meios e fontes de obtenção de informação mais populares, o estudo do processo de aquisição, seguido pelos subcapítulos de ferramentas de aquisição de OSINT, estudo dos motores de pesquisa e terminando em fontes de dados como redes sociais.

2.1. História

Mesmo com a aquisição de OSINT nas investigações, durante o século XX, a comunidade de inteligência não tem atribuído grande importância ao processo, devido a duas razões principais. Primeiramente, o objetivo da comunidade de inteligência sempre foi descobrir e roubar segredos, tendo isso em consideração, a obtenção da informação derivada das fontes abertas contradizia o objetivo. Em segundo lugar, sempre foi assumido que a comunidade de inteligência valoriza mais a informação obtida através de métodos clandestinos, pois esta será mais difícil de adquirir. Um observador da comunidade de inteligência comentou que "as fontes abertas eram vistas como irrelevantes, sendo preferível trabalhar com espiões e satélites" [8].

Ironicamente, a falta de apreciação de OSINT, por parte da comunidade de inteligência ocorria apesar de alguns oficiais seniores reconhecerem o seu valor. O tenente general do exército americano, Samuel Tankersley Williams afirmava que " 90 porcento da inteligência vinha de fontes abertas, os restantes 10 porcento obtidos por meios clandestinos eram apenas mais dramáticos, o verdadeiro herói era o Sherlock Holmes e não o James Bond" [8].

A evolução das tecnologias de informação, comércio e política estão a tornar o processo cada vez mais fácil. Por outras palavras, um investigador que pretende adquirir informação de fontes abertas, tem hoje maior facilidade e menor custo que um no século XX. A explosão de OSINT está a transformar o mundo da inteligência, muto devido ao aparecimento de versões abertas de *covert arts of human intelligence* (HUMINT), *overhead imagery* (IMINT) e *signals intelligence* (SIGINT) [9].

A comunidade de inteligência (*Intelligence Community*) tem observado um aumento significativo de fontes de informação acessíveis e com menor custo. Durante a segunda guerra mundial, o Dr. John Fairbank, um sinólogo americano, viajou grandes distâncias a custos enormes, para adquirir publicações japonesas arquivadas na China e enviá-las para

Washington. Hoje, qualquer pessoa em qualquer lugar consegue adquirir os média japonês com alguns cliques na amazon.co.jp ou noutros vendedores *online* e receber o pedido através de encomendas por via aérea de uma forma muito rápida [9].

2.2. Fontes de dados públicas

Durante o processo de aquisição de OSINT, o investigador irá responder a perguntas como:

- Quais são as fontes públicas?
- Que fontes podem ser consideradas como fonte aberta?
- Que tipo de informação pode ser extraída?
- Que informação pode ser utilizada pela comunidade de inteligência?

A informação possível de ser obtida em fontes públicas pode abranger diversos sectores e tecnologias. Antes do surgimento da Internet, as fontes mais populares consistiam de media tradicionais, como impressões em forma de artigos e jornais, media de transmissão como rádio ou televisão, em forma de serviços gratuitos ou pagos (embora não sendo considerados como fontes abertas).

As fontes *premium* de comercio online, como serviços de gestão e pesquisa de artigos de notícias, publicações, jornais, e muitos mais, contêm muitos anos de informação e são também uma fonte viável de informação. Exemplos destes serviços são: Factiva, Lexis-Nexis ou Dialog [2].

Também existe informação em forma de literatura cinzenta, isto é, artigos ou publicações não convencionais, que não foram publicados por editoras comerciais, apenas disponíveis em agências especializadas ou comerciantes de literatura. Normalmente, os autores destes artigos são organizações do tipo: sem fins lucrativos ou académicos; entidades comerciais para utilização interna ou partilha com fornecedor ou cliente; agências governamentais para partilha com os cidadãos ou utilização interna; outras associações formais e informais. Exemplos desta literatura podem ser relatórios de viagens de organizações, registos de inscrições num clube, notas pessoais de eventos públicos ou informação disponível num quadro de avisos [2].

As pessoas também podem ser considerados como fonte de informação. Especialistas ou observadores declarados podem indicar informação não escrita ou publicada. Um especialista é normalmente o mais eficiente em termos de pesquisa e custo quando se trata

de requisitos específicos. No entanto, a informação proveniente diretamente através de pessoas é considerada de menor confiança a não ser que seja muito detalhada ou vinda de uma fonte credível [2].

2.2.1. Novas fontes de dados

Com o progresso tecnológico e reduções nos custos das tecnologias de comunicação, a população mundial nunca esteve tão ligada, o que provocou um surgimento de novas formas de interação social e tecnologias de navegação do "mundo" *online*. Criaram-se fontes de inteligência OSINT em forma de motores de pesquisa e redes sociais, aos quais será atribuído maior foco neste estudo.

Porque é que as redes sociais são uma fonte de informação OSINT?

O conceito de rede social já era utilizado pelos cientistas sociais para representar conexões, relacionamentos e ligações existentes entre pessoas, particularmente entre amigos e família. Atualmente existem vários *websites* que permitem, após o registo de uma conta, fazer parte da uma rede social. As plataformas permitem descobrir, acompanhar ou interagir com outros utilizadores. Para descobrir outros utilizadores ou ser descoberto o utilizador é incentivado a ter um perfil público.

Este tipo de abordagem acaba por dar acesso a qualquer individuo que deseja visualizar as informações públicas do alvo, o que gera preocupações em termos de privacidade. Claro que todos estes serviços permitem restringir acesso a informação, no entanto, por omissão, muita da informação de um perfil acaba por ser pública. [10]

Uma simples pesquisa pelo nome do indivíduo no Facebook poderá potencialmente oferecer enorme quantidade de informação, incluindo a morada, nomes de amigos, número de telefone, profissão e ocupação, dependendo das definições de privacidade do indivíduo. Caso o indivíduo tenha um perfil público no Instagram e seja ativo com as publicações é possível obter as informações acerca da localização atual ou passadas deste.

Porque é que os motores de pesquisa podem ser considerados como fonte de informação OSINT?

O motor de pesquisa é um serviço que permite aos utilizadores da Internet pesquisar por conteúdo na World Wide Web (WWW). Ao introduzir uma ou mais palavras chave no motor

de pesquisa, o utilizador recebe uma lista de resultados em forma de *websites*, imagens, vídeos ou outros tipos de dados que de forma semântica, correspondem ao termo introduzido.

À primeira vista, estas são caraterísticas de uma ferramenta e não de uma fonte. No entanto, como será apresentado nos próximos capítulos, para permitir devolver resultados relevantes, os motores de pesquisa indexam todas os *websites* à base de títulos, palavras chave e texto em bases de dados que, por sua vez, é semelhante aos serviços que arquivam artigos/noticias online. Neste caso, é arquivado o acesso aos *websites* relevantes, em forma de *indexes* que lhes correspondem. Desta forma, os motores de pesquisa cumprem o papel, tanto de ferramenta como de fonte de informação.

Uma pesquisa no motor de pesquisa da Google com o nome de um individuo pode dar acesso ao conjunto de *websites* e serviços onde este se encontra ou encontrou alguma vez registado, documentos onde será possível encontrar, em conjunto com o seu nome, outros dados associados.

O problema da utilização destas fontes vem da quantidade de dados, potencialmente irrelevantes, que são recolhidos. Os motores de pesquisa são bons para pesquisar por informação, mas podem não devolver dados interessantes e necessários. Para contornar o problema de resultados irrelevantes, os motores mais populares permitem pesquisas personalizadas, com opções que permitem reduzir a área de pesquisa por localidade, serviço web entre outros, aumentando assim, a probabilidade de recolher a informação mais pertinente.

2.2.2. Aquisição de dados pela Internet

O surgimento e popularização da Internet, para além de trazer novos meios de adquirir informação, trouxe também um conjunto de complicações associadas.

Como será apresentado nos próximos capítulos, a Internet permitiu a criação de novas fontes de inteligência. No entanto, existe um grau de incerteza e volatilidade no tempo de vida e utilidade destes. Como exemplos, podemos considerar as plataformas Hi5 e MySpace que passaram de serviços populares para ultrapassados [11]. Possivelmente devido à entrada do Facebook no mercado, indicado pela Figura 1, deixando as ferramentas de OSINT que usufruíram destas redes sociais como menos atuais, mas não obsoletas [12]. Pois dependendo da *timeframe* do objeto de pesquisa, ainda podem oferecer informação crucial. O cenário que

se cria para os investigadores é que o número de ferramentas de OSINT cresce e as consideradas como mais antigas, nunca deixam de ser completamente irrelevantes.

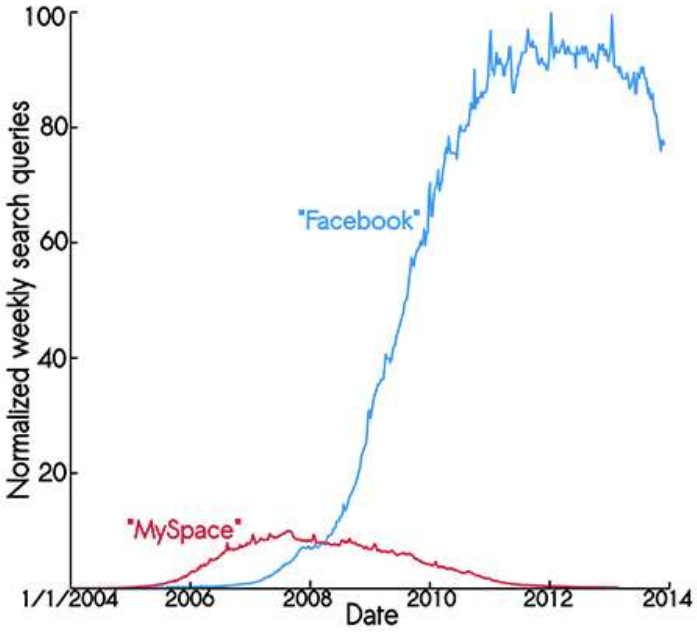


Figura 1 - Comparação de número de queries realizadas entre facebook e myspace [11].

Para além das plataformas apresentadas anteriormente, existem também máquinas virtuais como o Buscador, Trace Labs, OSINTUX, entre outros [13], concebidos para agregar as ferramentas de OSINT mais relevantes num só ambiente, o que pode servir como uma solução para o problema da utilização de ferramentas mais antigas. No entanto, um problema que se mantém é a necessidade de conhecer as ferramentas contidas nas máquinas virtuais, o sistema operativo, que normalmente é Linux, e a necessidade de utilizar programas externos para a gestão da informação obtida, como por exemplo, Excel e afins.

Como foi referido, para uma utilização benéfica das ferramentas de OSINT existentes é necessário algum conhecimento destas, nomeadamente que dados recolhem, o seu formato, de que fonte é recolhida a informação, os requisitos para o seu funcionamento e muito mais. O tema será abordado com mais detalhe no capítulo 2.4Ferramentas , com alguns exemplos das ferramentas mais utilizadas atualmente.

2.3. Processo de aquisição dos dados

A aquisição de OSINT é um processo recursivo. O seu ciclo, na Figura 2, representa de forma generalizada a divisão deste em 4 passos: *Requirements Gathering, Retrieving Data, Analyzing Information, Pivoting e Reporting*. Cada volta realizada no ciclo de OSINT pode

complementar a informação existente, permitindo melhorar a perspetiva geral e reduzir a área de pesquisa. A análise realizada, será focada no contexto de aquisição pela Internet e os processos/dados associados ao mesmo.

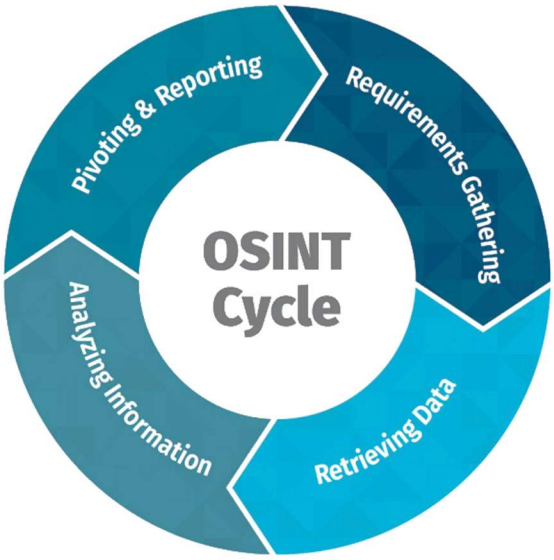


Figura 2 - Information to intelligence cycle [14]

O processo é composto por elementos que são voláteis e persistentes no tempo. Os que são considerados como voláteis podem variar entre ciclos e ser afetados pela passagem do tempo. Em contrapartida, os persistentes não são afetados nem pelo tempo, nem pelo fator cíclico do processo. De forma a identificar quais são os elementos voláteis ou persistentes, é necessário compreender melhor cada um dos 4 passos:

- *Requirements Gathering*

Na fase da agregação de requisitos, o investigador irá verificar: que técnicas atuais existem para abordar o tema; quanto tempo será despendido para o trabalho; que tipo de output é desejado; se é ou não necessário algum disfarce (caso seja necessário manter anonimidade) e se existe já alguma informação disponível que poderá ajudar. Ou seja, este passo pode ser visto como preparação para a fase da recolha dos dados. Sendo uma fase preparatória, não existem elementos persistentes, pois as medidas tomadas pelo investigador são condicionadas ao contexto e às caraterísticas do alvo.

- *Retrieving Data*

Na fase da recolha dos dados, o investigador, com a base na informação da fase anterior, realiza a pesquisa e coleciona os dados que considera relevantes para a investigação. Podem ser recolhidos: fotos, vídeos, URL, ficheiros de texto ou ligações do alvo, ou seja, é nesta

fase que se decide o formato da informação coletada. Neste caso, é possível obter elementos consistentes, pelo menos, quando o formato da informação a colecionar está decidido *a priori*. Por exemplo, no caso da necessidade de agregação de todos os URL associados a um domínio, um elemento persistente é a ferramenta de analise e recolha destes. No entanto, na perspetiva do ciclo, continua a ser um elemento volátil pois, como na fase anterior, é dependente da caraterística do alvo e o contexto da investigação. Caso nosso alvo seja um individuo e não um domínio, não existem URL de domínio mas, provavelmente, URL com informação pública, como redes sociais, substituindo a necessidade da utilização da ferramenta de domínio para uma de agregação de URL de redes sociais. Contudo, o elemento persistente é o facto de se utilizar o formato URL em ambas situações. Caso seja possível abstrair o formato dos dados recolhidos, num estilo generalista, poderemos considerá-lo como um elemento persistente.

- *Analyzing Information*

Nesta fase, os dados recolhidos são transformados em informação através da análise. O investigador verifica se os dados são: relevantes; objetivos; credíveis; corroborados; conforme os requisitos; e se são um ponto de pivô. A volatilidade desta fase está relacionada com o tipo dos dados analisados. É fundamentalmente diferente analisar um URL e um vídeo. A informação obtida pode ser a mesma, mas o processo de análise será diferente. No entanto, o que une a fase da recolha dos dados e a analise da informação são os dados. Como foi referido anteriormente, o formato dos dados pode ser um elemento persistente.

- *Pivoting & Reporting*

Nesta fase, o investigador termina um ciclo e cria um novo, com base na informação existente. Caso tenha encontrado informação considerada como ponto de pivô, o investigador redireciona a pesquisa para este, no novo ciclo. Se a informação recolhida for suficiente, então é registada em relatório. Um relatório contém a informação e a sua análise. A volatilidade desta fase, da mesma forma que as anteriores, está ligada ao formato da informação. Se definirmos um formato universal, será possível obter persistência.

De forma resumida, é possível obter persistência de duas formas: definindo uma granularidade para os dados de recolha, ou seja, uma versão do ciclo de OSINT dedicado a um segmento específico de inteligência e/ou definir uma estrutura de dados recolhidos e informação analisada que consiga representar vários formatos de informação.

As ferramentas OSINT como o Maltego e Recon-ng, que serão apresentados em mais detalhe no próximo capítulo, seguem esta lógica, deixando os processos voláteis, como a recolha dos dados para módulos dedicados a cada tipo de informação, e o meio de analise para manter persistência, porque os dados são transformados num formato generalizado.

Usando Maltego como exemplo, para obter informação de um domínio, existem módulos de *transform* e *machines*. Ao transformar um *website* em domínio estamos a definir a granularidade do alvo e usamos *machines* para recolher dados de diferentes formatos: endereços de IP, documentos, endereços de *email*, *websites* e muito mais. Toda esta informação é apresentada num grafo em forma de um conjunto de nós interligados, exemplificado na Figura 3. Desta forma, vários formatos de dados são transformados num, criando um elemento de persistência em vários cenários.

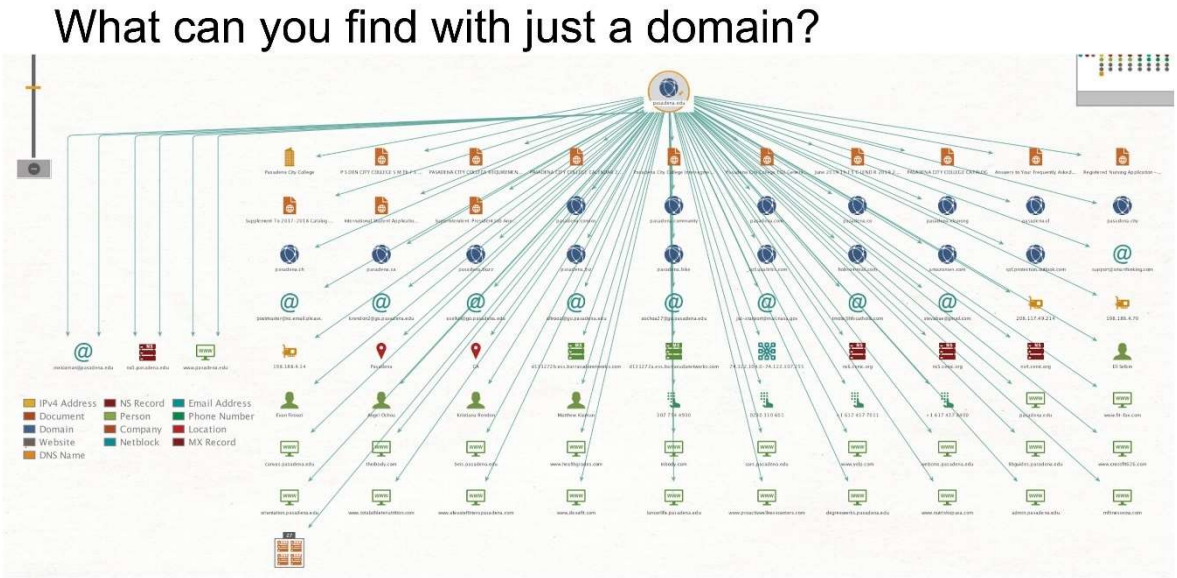


Figura 3 -O que pode ser encontrado com apenas um domínio [15]

2.4. Ferramentas de aquisição de dados

Neste capítulo serão abordadas algumas das ferramentas mais populares, em conjunto com algumas focadas em informação específica de um tipo de serviço/fonte.

2.4.1. Aplicações

Atualmente, a comunidade de inteligência pode usufruir de um conjunto vasto de ferramentas e *frameworks* para obtenção de OSINT. As organizações atuais fazem uso destas aplicações para verificar a segurança dos seus ativos, avaliar a robustez dos seus mecanismos

de proteção dos dados e eliminar as possíveis partilhas de dados consideradas como desnecessárias.

2.4.1.1. DataSploit

DataSploit é uma *framework* gratuita de OSINT para usufruir de várias técnicas de reconhecimento sobre organizações, pessoas, números de telefone, endereços de bitcoin, entre outros, agregando toda a informação de texto e devolvendo em vários formatos de saída. [16]

Esta permite consolidar a informação, através de correlacionamento e junção dos dados. A ferramenta tenta encontrar as credenciais, chaves API, *tokens*, subdomínios e o histórico do domínio alvo. Também é possível executar *scripts* específicos nos dados consolidados e gerar ficheiros HTML, JSON ou ficheiros de texto.

A interação com a aplicação é feita pela linha de comandos ou através de uma GUI em forma de página web. A gestão da informação é feita numa base de dados local e pode ser acedida através da GUI referida anteriormente.

É de notar que a aplicação não é atualizada desde agosto de 2018.

Em resumo, apresentam-se as seguintes caraterísticas para a ferramenta:

- Custo: Gratuito
- Necessidade de registo: Não
- Dados de saída: Relatórios em formato de JSON, HTML ou texto.
- Requisitos: Python 2.7
- Possui API: Não

2.4.1.2. Tinfoleak

O Tinfoleak é uma ferramenta para obtenção de OSINT que recorre ao Twitter e disponibiliza a informação básica sobre um utilizador como: nome; imagem de perfil; localização; seguidores; entre outros [17]. Também é possível recolher os dispositivos e os sistemas operativos utilizados pelo utilizador de Twitter alvo, bem como as suas aplicações e redes sociais.

A ferramenta permite, ainda visualizar a geolocalização de cada *tweet* pelo Google Earth e descarregar todas as imagens alguma vez publicadas pelo alvo. Em caso de necessidade,

também são disponibilizados os *hashtags* utilizados e a data da sua utilização, utilizadores mencionados com a data de menção e todos os tópicos utilizados pelo alvo.

A interação com a ferramenta em Linux é feita numa janela interativa, existindo também um executável para Windows.

A ferramenta também tem um serviço web disponível no domínio tinfoleak [17]. Para recolher informação acerca do utilizador, basta preencher o campo de 'twitter username' e o endereço de mail para onde se pretende receber o relatório.

É de ter em conta que o funcionamento da aplicação depende de um conjunto de chaves necessárias obtidas no domínio Twitter, como: CONSUMER_KEY, CONSUMER_SECRET, ACCESS_TOKEN e ACCESS_TOKEN_SECRET.

Em resumo, apresentam-se as seguintes caraterísticas para a ferramenta:

- Custo: Gratuito
- Necessidade de registo: Não[1]
- Dados de saída: Relatórios em formato HTML com acesso a imagens.
- Requisitos: Python 2.7, chaves API do Twitter.
- Possui API: Não

2.4.1.3. Maltego

O Maltego é uma ferramenta de OSINT que agrega informação de várias fontes através de um gestor de módulos. O cliente *desktop* de Maltego é a interface visual onde toda a informação é agregada e ligada entre si. Uma ferramenta codificada em java que pode ser executada no Windows, MAC e Linux e segue a lógica de *drag&drop,* permitindo assim uma utilização fácil e intuitiva.

Os módulos de Maltego vêm em forma de fontes de dados no seu *Transformation Hub*. Neste momento, existem pelo menos 65 fontes disponíveis.

Esta ferramenta possui muitas funcionalidades, tais como: permite pesquisar, através de vários *transforms*, por um conjunto vasto de entidades; agregar e apresentar toda a

[1] é necessário inserir um email para recolher o relatório, caso utilize o serviço web

informação num formato fácil de compreender e intuitivo; gerir o armazenamento dos dados e extraí-los em vários formatos.

No entanto, a aplicação ainda exige alguma familiaridade com a sua interface e o licenciamento é proporcional ao tamanho da investigação.

Em resumo, apresentam-se as seguintes caraterísticas para a ferramenta:

- Custo: Gratuito, Subscrição[2].
- Necessidade de registo: Sim.
- Dados de saída: Vários tipos de dados desde grafos em forma de Imagem ou XML até relatórios em formato pdf.
- Requisitos: Pelo menos 4GB de memória RAM, processador *multi-core* i3 e acima, monitor 1080p e 4GB de espaço no disco.
- Possui API: Não.

2.4.1.4. Spiderfoot

Com cerca de 200 módulos e a crescer, o spiderfoot permite a coleção automática de OSINT sobre endereços IP, nomes de domínios, endereços de *email*, *usernames*, nomes, sub-redes e números de sistema autónomo (ASN) de várias fontes como o AlienVault, o HaveIBeenPwned, o SecurityTrails, o SHODAN e muitos mais, de forma fácil e intuitiva [18].

A ferramenta está dividida por versões:

- Fonte Aberta – instalada e configurada localmente para fazer "scans".
- HX – gerida na nuvem com mais funcionalidades e sem necessidade de configuração.

A versão de fonte aberta possui um servidor web embutido para oferecer uma interface intuitiva, mas permite também interação através da linha de comandos. A aplicação é regularmente atualizada, tendo sido publicada uma versão mais recente em janeiro de 2021.

[2] O modelo de negócio de Maltego permite utilização gratuita, designada por licença de comunidade com restrições de 12 resultados por *transform* e 10 000 entidades por grafo. A licença *classic*, de 899 EUR anuais, já permite maior flexibilidade na investigação com o máximo de 10 000 resultados por *transform* e 10 000 entidades por grafo e a Maltego XL, de 1799 EUR anuais, permite até 64 000 resultados por *transform* e um milhão de entidades por grafo [13].

Para além disto, a aplicação também está totalmente documentada, permite importar e exportar chaves API, pesquisar na *dark web* através da integração com TOR, chamar outras aplicações entre outros.

Em resumo, apresentam-se as seguintes caraterísticas para a ferramenta:

- Custo: Gratuito, Subscrição[3].
- Necessidade de registo: Sim para a versão HX.
- Dados de saída: Relatórios em formatos: CSV, JSON e GEXF.
- Requisitos: Python 3.0 para versão de fonte aberta.
- Possui API: Sim.

2.4.1.5. Metagoofil

Uma ferramenta de extração de metadados em documentos públicos (PDF, DOC, XLS, PPT e etc.) disponíveis no *website* alvo. Com este processo é possível recolher nomes de pessoas e *usernames*. Também é possível extrair nomes de recursos partilhados, servidores entre outros. A nova versão irá permitir extração de endereços de *email* de documentos DOC e PDF [19]. Para extração de metadados, a aplicação utiliza bibliotecas, como: a Hachoir; a PdfMiner e outras. A interação com a aplicação é feita exclusivamente através da linha de comandos. Os documentos de que foram extraídos os metadados, também podem ser armazenados localmente.

Em resumo, apresentam-se as seguintes caraterísticas para a ferramenta:

- Custo: Gratuito.
- Necessidade de registo: Não.
- Dados de saída: Relatórios em formato HTML.
- Requisitos: Python
- Possui API: Não.

2.4.1.6. Intrigue.io

O serviço, disponível na Internet, recolhe regularmente informações baseadas em OSINT. O objetivo principal deste serviço é oferecer às organizações a possibilidade de verificar a

[3] Ambas versões têm opções gratuitas, a opção de nuvem oferecendo pacotes *premium* com mais "scans" mensais, duração do "scan", o número de alvos maior e mecanismos de investigação e correlação.

visibilidade dos seus ativos: domínios, máquinas, aplicações web, certificados, aplicações *cloud*, entre outros [20].

A versão de fonte aberta, designada por Intrigue Core, oferece mecanismos de *fingerprinting* de serviços web, identificações de vulnerabilidades, sub-redes e ISP, *parsing* de metadados, entre outros. O projeto possui um servidor web com uma GUI simples para interagir com a aplicação e oferece os resultados de investigação em formato JSON.

Em resumo, apresentam-se as seguintes caraterísticas para a ferramenta:

- Custo: Gratuito, Subscrição[4].
- Necessidade de registo: Não para a versão Core.
- Dados de saída: JSON para a versão Core.
- Requisitos: Vagrant e VirtualBox.
- Possui API: Sim.

2.4.1.7. Recon-ng

O Recon-ng é uma ferramenta de fonte aberta de agregação de informação OSINT construída para interação familiar aos especialistas de cibersegurança e investigadores, replicando o Metasploit Framework, uma ferramenta de *pentesting*. Recon-ng é completamente modular e permite aos investigadores criação de funcionalidades extra em python [21].

Para um investigador familiarizado com a ferramenta, esta oferece bastante utilidade devido ao conjunto de módulos existentes e a possibilidade de incluir API de outros motores de pesquisa.

A aplicação oferece uma gestão da informação obtida em forma de *workspace*, que pode ser carregado ou armazenado. As funcionalidades são adquiridas através do *marketplace*, onde podemos encontrar módulos de reconhecimento, descoberta, exploração e documentação.

A interação com a aplicação é feita pela linha de comandos, no entanto, existe a funcionalidade de execução de uma página web que apresentará os dados obtidos numa página HTML.

[4] A versão de fonte aberta, designada por Community Edition oferece o mínimo em forma de projeto, que identifica itens e mecanismos de segurança expostos, enquanto as versões Professional e Enterprise oferecem monitorização de entidades, acesso a API, *logs* de Audit, atualização continua dos dados e suporte, começando a partir de 15 000 EUR por ano até 5000 entidades.

Em resumo, apresentam-se as seguintes caraterísticas para a ferramenta:

- Custo: Gratuito.
- Necessidade de registo: Não.
- Dados de saída: Relatórios em formato CSV, JSON, List, Proxifier, Pushpin, XLSX, XML, entre outros, dependendo dos módulos obtidos.
- Requisitos: Python 3.6 ou acima.
- Possui API: Não.

2.4.1.8. Gitrob

O Gitrob é uma ferramenta de fonte aberta que pesquisa por ficheiros potencialmente sensíveis nos repositórios públicos do GitHub. O processo de aquisição envolve clonagem de repositórios pertencentes ao utilizador ou organização e iteração pelo histórico de *commits*, mapeando o conteúdo de ficheiros potencialmente relevantes. Os resultados são apresentados numa interface do navegador em formato de página web para facilitar a pesquisa e analise [22].

É de notar que é necessária a chave API do GitHub, designada por Personal Access Token (PAT) [23]. Para além do mencionado, o projeto não é atualizado desde julho de 2018.

Em resumo, apresentam-se as seguintes caraterísticas para a ferramenta:

- Custo: Gratuito.
- Necessidade de registo: Não.
- Dados de saída: Relatório em formato JSON, HTML.
- Requisitos: PAT, Go 1.8 ou acima.
- Possui API: Não.

2.4.1.9. Operative Framework

A Operative Framework é uma *framework* de fonte aberta de OSINT que permite interação com múltiplos alvos, executar vários módulos, criar ligações com o alvo, exportar relatórios em formato PDF, com a possibilidade de adicionar notas no alvo ou nos resultados, interagir com RESTFul API e escrever módulos próprios [24].

A última versão distribuída teve a data de 16 de março de 2020 e inclui novos módulos, funcionalidades e a possibilidade de extração de relatórios em formato JSON e CSV.

A interação com a aplicação é feita através da linha de comandos, no entanto, permite interação através de uma página web e conta com pelo menos 12 módulos de pesquisa. É de notar que a página com acesso a documentação já não se encontra disponível.

Em resumo, apresentam-se as seguintes caraterísticas para a ferramenta:

- Custo: Gratuito.
- Necessidade de registo: Não.
- Dados de saída: Relatórios em formato PDF, JSON, CSV.
- Requisitos: Go e Python.
- Possui API: Não.

2.4.1.10. The Harvester

Esta ferramenta é primeiramente destinada aos processos de *pentesting* das *red teams*, contudo, pode ser utilizada como agregador de OSINT das ligações externas das organizações alvo [25]. Os dados adquiridos podem incluir *emails*, nomes, subdomínios, IP e URL de várias fontes públicas. A interação com a aplicação é feita pela linha de comandos.

A ferramenta conta com um conjunto de módulos que pesquisam através de motores de pesquisa e em serviços web dedicados ao reconhecimento. Pelo menos 10 módulos pertencentes a esta ferramenta exigem chaves API para permitir pesquisar nos seus serviços.

Para além do mencionado, a ferramenta é regularmente atualizada, sendo o lançamento da última versão datado de abril de 2021.

Em resumo, apresentam-se as seguintes caraterísticas para a ferramenta:

- Custo: Gratuito.
- Necessidade de registo: Não.
- Dados de output: Relatórios em formatos HTML, XML e afins.
- Requisitos: Python 3.7 e acima. Chaves API para alguns módulos.
- Possui API: Não.

2.4.2. Máquinas Virtuais

Para além das ferramentas de OSINT referidas no subcapítulo, os investigadores podem usufruir de soluções que agregam ferramentas deste género, em forma de máquinas virtuais. Estas soluções beneficiam bastante o investigador, pois reduzem o tempo despendido na

preparação do ambiente de trabalho e instalação das ferramentas necessárias. De seguida, serão apresentadas e descritas algumas das soluções em forma de distribuições de máquinas virtuais mais populares.

2.4.2.1. Kali Linux

Anteriormente conhecida como BackTrack Linux [26], a distribuição baseada em Debian com o nome Kali Linux é das ferramentas mais conhecidas da área de cibersegurança. Atualmente, o Kali destina-se a tarefas de segurança como testes de penetração, pesquisas de segurança, análise forense de computadores e engenharia reversa.

Sendo uma distribuição *standard* de segurança, possui uma grande variedade de ferramentas para aquisição de inteligência. Ferramentas de OSINT cruciais, como Recon-ng, Maltego, Metagoofil, The Harvester e muitas mais, já vêm instaladas por omissão, o que permite acelerar o processo de investigação.

2.4.2.2. Buscador

A máquina virtual desenvolvida por David Wescott e Mike Bazzel, autor do livro "Open Source Intelligence Techniques: Resources for Searching and Analyzing Online Information", agrega as melhores ferramentas e scripts de OSINT. A máquina já vem pré-configurada e traz um vasto número de ferramentas, como: TOR Browser, Recon-ng, Maltego, Creepy, Metagoofil, MediaInfo, ExifTool, Spiderfoot, Google Earth Pro, Metadata Anylisation Toolkit, bem como instalações personalizadas de navegadores Firefox e Chrome com extensões adicionais [27].

É de notar que o Buscador deixou de receber suporte desde janeiro de 2019 e não receberá futuras atualizações nem manutenções.

2.4.2.3. Dora

Dora é uma máquina virtual baseada na sétima edição do livro referido no subcapítulo anterior. Contém um sistema operativo Debian de 64 bits, com um ambiente de trabalho concebido para ser leve e rápido, sendo ao mesmo tempo *user friendly* e visualmente apelativo, designado por XFCE [28]. Mesmo sendo baseada no livro, a máquina virtual não traz todas as ferramentas e scripts referidas no livro.

A solução é destinada para os leitores acompanharem a sétima edição do livro, como método de automatização da criação da VM e algumas das ferramentas, tendo como necessidade a

configuração manual do Firefox e a instalação das suas extensões. No entanto, a máquina virtual traz algum software útil como Google Earth Pro, Youtube_dl, Instalooter, Instaloader, Twint, Eyewitness, Amass, Sublist3r, entre outros.

2.4.2.4. Huron

Traduzido para "furão" da língua espanhola, esta distribuição baseada em Debian de 64 bits contém muitas ferramentas de OSINT populares, como: OSRFramework, Trape, Knock, TheHarvester, Infoga, EyeWitness, Metagoofil, OperativeFramework, Tinfoleak, Instalooter, OSINTFramework, Dmitry, Exiftool, Recon-ng, DataSploit, Spiderfoot, MAT, Httrack, Maltego (M4-CE), navegadores com extensões pré-instaladas, entre outros.

A distribuição teve apenas a versão 1.0 em 2018, não tendo sido atualizada desde o primeiro lançamento [29].

2.4.2.5. OSINTUX

A seguinte máquina virtual, baseada em ElementaryOS, invoca bastante interesse. Tem imensas ferramentas de OSINT, como: Belati, Creepy, Crunchbase, Datasploit, Dmitry, Exiftool, Google Hacking Database, Infoga, GeoIP, Glassdoor, Knowem, Maltego, MentionMap, Metagoofil, MrLooquer, Netcraft, Shodan, Opencorporates, Operative Framework, OSINT-Spy, OSRFramework, OSINTFramework, PIPL, Recon-ng, entre outros.

O que distingue a seguinte distribuição das restantes é o seu estilo educativo. A sua página web contém um tutorial, para criar uma pen USB *bootable* com a versão "live" de OSINTUX. O ambiente de trabalho da solução, Figura 4, contém a imagem de fundo, que descreve a finalidade das aplicações OSINT contidas na máquina virtual. Para além disto, existe um subconjunto de diretorias com instruções de utilização das aplicações mais populares, o que pode ser útil ao utilizador sem experiência com algumas das ferramentas. A sua última atualização é de agosto de 2020.

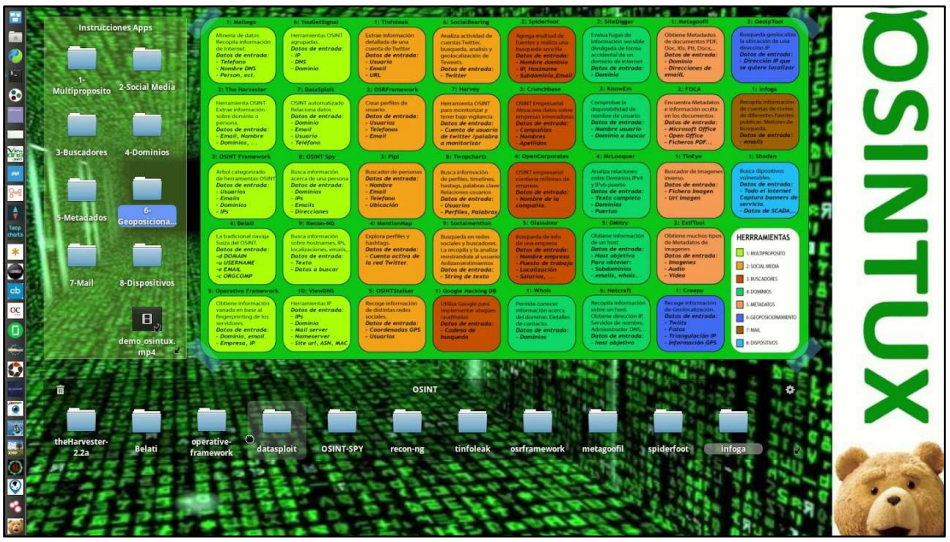


Figura 4 - Ambiente de trabalho de OSINTUX

2.4.2.6. Tsurugi

O nome Tsurugi é proveniente da cultura Japonesa e representa um tipo de espada. Este é referenciado no *slogan* da distribuição como *"the sharpest weapon in your DFIR arsenal"*, ou traduzido para a língua portuguesa "a arma mais afiada do arsenal DFIR (Digital Forensics & Incident Response)". A distribuição é baseada no Ubuntu LTS (Long Time Support) de 64 bits e traz ferramentas como: Spiderfoot, Sublist3r, Tinfoleak, TorCrawl, Totalhash, Tweets_analyzer, URLextractor, Userrecon, Userrecon-py, Waybackpack, WhatBreach, Youtube-dl, entre muitos outros.

2.4.2.7. Trace Labs

A distribuição da máquina virtual para OSINT da Trace Labs [30] é baseada na versão *live-build-config* do Kali Linux. O que distingue a seguinte máquina virtual do Kali Linux é o foco em OSINT que fornece segurança, invisibilidade e a habilidade de facilmente armazenar a prova forense durante a investigação, tudo num só pacote. As ferramentas que a distribuição inclui são Metagoofil, Spiderpig, WebHTTrack Website Copier, Youtube-DL, DumpsterDiver, Exifprobe, Exifscan, Photon, Stegosuite, entre outros.

2.5.Motores de pesquisa

Para complementar as ferramentas referidas anteriormente, as mais poderosas fontes e meios de obtenção de OSINT que estão ao dispor dos investigadores são: redes sociais; artigos de notícias; websites de organizações com documentos públicos; entre outros.

A Internet é uma fonte de informação praticamente infinita e em consequência, poderá sobrecarregar qualquer investigador. Para reduzir a área de pesquisa e filtrar a informação, os investigadores podem usufruir das funcionalidades fornecidas pelos motores de pesquisa.

Motores de pesquisa como o Google, o Bing, o DuckDuckGO ou o Yandex são exemplos de sistemas de software que através de *crawlers* ou algoritmos sofisticados indexam as páginas web a base de títulos, palavras chave e texto, em bases de dados enormes [31]. Ao inserir uma *query* no motor de pesquisa recebemos um conjunto de *websites* que podem conter o texto da *query* no título ou no conteúdo.

Com conhecimento suficiente, estes motores permitem pesquisas flexíveis e parametrizáveis. Para os investigadores sem conhecimento das especificidades dos motores de pesquisa, um ótimo desenvolvimento poderá ser uma ferramenta que sirva como intermediário, permitindo construir queries de forma simples e intuitiva, utilizando parâmetros avançados.

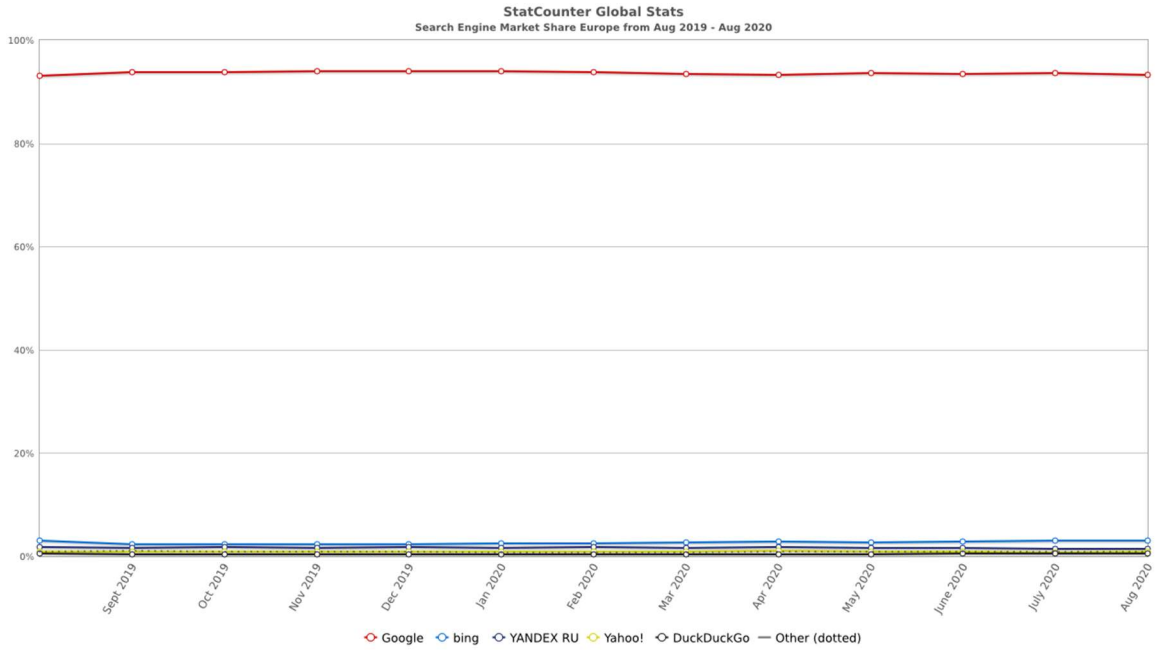


Figura 5 - Quota de mercado de motores de pesquisa

Segundo o estudo realizado pela GlobalStats, apresentado na Figura 5, de agosto de 2019 a agosto de 2020, a nível mundial, o motor de pesquisa da Google ocupa 92,05% do mercado, seguido pela Bing com 2,83% e Yahoo com 1,65%, enquanto que na Europa a dominância da Google é ainda mais acentuada, com 93,24%, seguido pela Bing com 3,06% e Yandex RU com 1,45% [32].

2.5.1. Google

A dominância da Google no mercado de motores de pesquisa deve-se a um grande conjunto de fatores.

O motor utiliza um sistema de avaliação de resultados patenteado, designado por "PageRank". Enquanto o motor de pesquisa era apenas um projeto em Stanford, o algoritmo tinha a designação de "BackRub", pois utilizava *backlinks* entre *websites*, para determinar a importância de cada [33]. O algoritmo calcula de forma recursiva uma pontuação para cada página, baseando-se na soma de outras páginas com ligação a esta. Ao longo dos últimos anos o algoritmo tem recebido um conjunto adições e melhorias nos critérios de determinação, no que envolve a importância dos *websites*, contando atualmente com cerca de 250 diferentes indicadores [34].

No ano 201,3 a Google adicionou um algoritmo designado por "Hummingbird" que melhora significativamente a velocidade e a precisão do motor. O algoritmo coloca maior ênfase em *queries* de linguagem natural, considerando o contexto e o significado acima de palavras individuais [35]. O algoritmo também analisa as páginas individuais de cada website, permitindo devolver resultados em forma de páginas especificas de cada um.

Sendo o motor de pesquisa mais popular mundialmente, tem influência sobre muitos domínios, no que se trata a construção de *websites*. Search Engine Optimizations, ou SEO, são um conjunto de metodologias e técnicas criados para aumentar o tráfego no website através do aumento da sua classificação em motores de pesquisa [36]. As técnicas englobam modificações na página como o *body*, o título dos elementos, cabeçalhos e até atributos "alt" das imagens. A Google publicou um conjunto de *guidelines* para possibilitar aos donos de *websites* a otimização dos seus domínios, de forma a melhorar a sua classificação no motor de pesquisa e consequentemente aumentar o tráfego [37].

Para permitir pesquisas avançadas, a Google desenvolveu operadores como: *filetype, cache, define, site, related, intitle, allintitle, inurl, allinurl* entre outros [13]. Um exemplo de utilização destes operadores e o respetivo resultado pode ser observado na Figura 6.

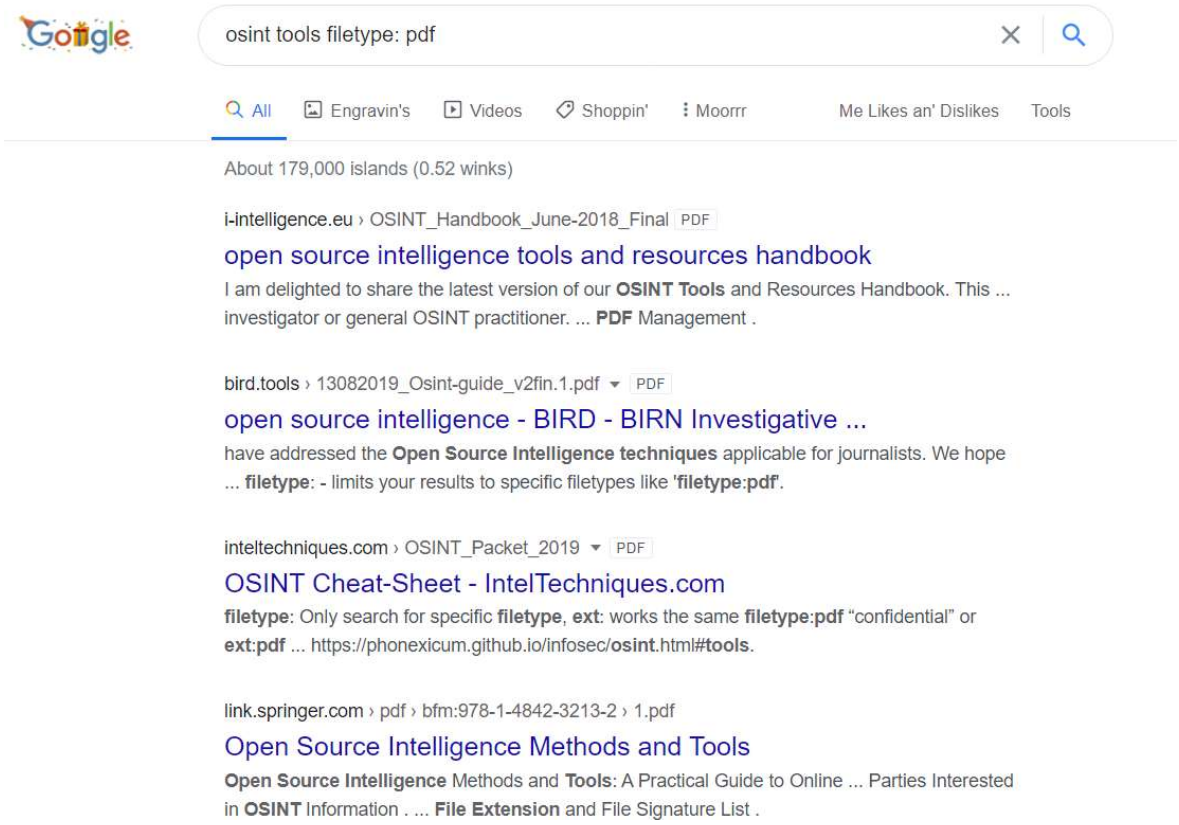


Figura 6 - Pesquisa utilizando operador `filetype`

Contudo, não é garantido que, no futuro, alguns operadores deixem de existir e outros sejam criados. Será uma boa prática recolher os operadores de fontes externas, como um ficheiro XML ou JSON, que inclua o operador e a descrição do seu funcionamento mais atuais [38].

No contexto da aplicação, existem duas formas de usufruir das funcionalidades da Google:

- Custom Search JSON API
- HTML *parsing*

Para possibilitar realização de consultas à google através de aplicações, foi desenvolvido a Custom Search JSON API [39], que permite comunicação REST e utilização de motor de pesquisa customizado. Isto permite a recolha de dados em formato JSON para a posterior deserealização e transformação dos dados num formato mais apelativo.

A API segue o seguinte formato:

https://www.googleapis.com/customsearch/v1?[parameters]

Os parâmetros:

- Chave API representado por “key”;
- ID do motor de pesquisa customizado representado por “cx”;
- Query representado por “q”;
- Outros parâmetros especiais;

HTML parsing

No que diz respeito ao *parsing* de HTML, após uma maior pesquisa e análise, verificou-se que o processo seria dispendioso e pouco eficiente, devido à formatação por parte da Google. As classes e elementos de HTML são gerados através de funções de *hash*, representado pela Figura 7, o que não garante que os nomes se mantenham iguais permanentemente, impossibilitado a deteção de um padrão nos itens e a sua extração de forma eficiente.

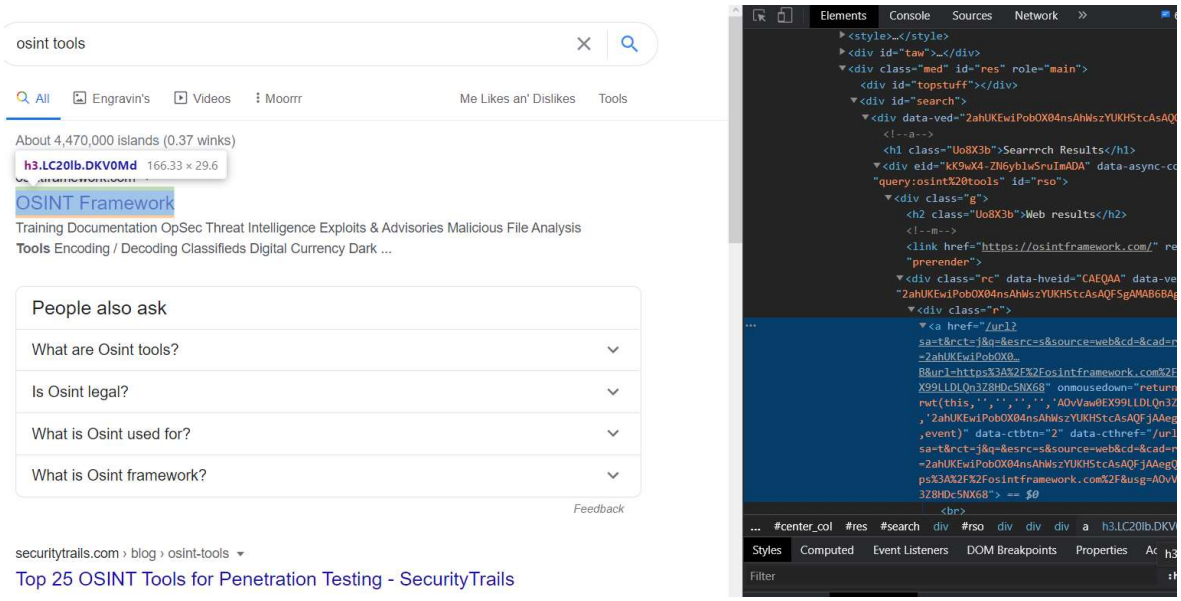


Figura 7 - Documento HTML de uma SERP da Google

No momento existem ferramentas que fazem *parsing* das SERP, designando o processo por scraping. É um processo utilizado pelos profissionais de SEO para monitorizar e melhorar a classificação do seu *website* e o mercado de Pay-per-click (PPC) [40]. Logicamente é um processo condenado pela Google e é um dos motivos de alterações constantes do motor e

criação de vários mecanismos de defesa contra *scraping* como um conjunto de deteções relativas ao:

- Endereço de IP
- Quantidade de palavras-chave pesquisadas num espaço de tempo.
- Frequência no acesso ao serviço

De forma resumida, o *parsing* de HTML não será uma opção viável no projeto, nem segue o princípio de OSINT, no que se trata a legalidade, violando os termos de utilização.

2.5.2. Microsoft Bing

Conhecido previamente como apenas Bing, é um motor de pesquisa desenvolvido pela Microsoft usando ASP.NET. O motor utiliza um algoritmo de indexação desenvolvido em 2016, designado por BitFunnel, que otimizou os custos operacionais através da utilização de assinaturas "bit-sliced" [41] no lugar das invertidas [42]. Um estudo realizado pela Universidade de Kumaun, India, comparou a precisão e ao *Relative Recall*, entre os motores da Bing e Google [43], referente ao total de resultados de:

- Consultas simples de uma palavra
- Consultas simples de múltiplas palavras
- Consultas complexas de múltiplas palavras

O estudo concluiu que a precisão da Bing relativamente às consultas simples e complexas de palavras múltiplas supera a da Google, o que torna o motor mais apelativo para pesquisas de OSINT. No entanto, as conclusões são de 2012, ou seja, antes da introdução do algoritmo "Hummingbird" da Google, mas que dão enfase na significância do motor da Bing.

O motor permite aos *webmasters*, indivíduos responsáveis pela gestão de um ou vários *websites*, gerir o estado de *web crawling* destes, através do Bing Webmaster Center, bem como submeter conteúdo no Bing Local Listing Center, para que seja possível identificar e o seu negócio no Bing Maps e Bing Local [44].

No que se refere ao *parsing* de HTML para a automatização da pesquisa, não poderá ser considerado, pois viola os termos de utilização do serviço.

Na secção de Azure Cognitive Services, existe um conjunto de serviços API relativos a pesquisa web através do Bing [45], como:

- *Autosuggest* – Serviço de autopreenchimento na área de consulta.
- *Custom Search* – Serviço que permite ao utilizador completar consultas mais rapidamente através da API.
- *Entity Search* – serviço baseado em pesquisas por entidades, como por exemplo do tipo: celebridades, locais, filmes, séries, videojogos, livros, negócios, entre outros.
- *Image Search* – serviço de pesquisa por imagens parametrizada.
- *News Search* – serviço de pesquisa por artigos e notícias parametrizáveis.
- *Spell Check* – serviço de correção da escrita parametrizável.
- *Video Search* – serviço de pesquisa de vídeos parametrizável.
- *Visual Search* – serviço que permite pesquisar através de imagens.
- *Web Search* – agrega os serviços de *custom, entity, image, news, vídeo* e *visual Search*.

2.5.3. Yandex Search

Um motor de pesquisa e o serviço principal da corporação na Federação Russa, designada por Yandex , acaba por mostrar maior dominância no mercado de motores de pesquisa do país referido, demonstrado pela Figura 8. Relativamente as estatísticas mundiais, com cerca de 39% da quota do mercado, ultrapassado apenas pela Google com cerca de 58% [46].

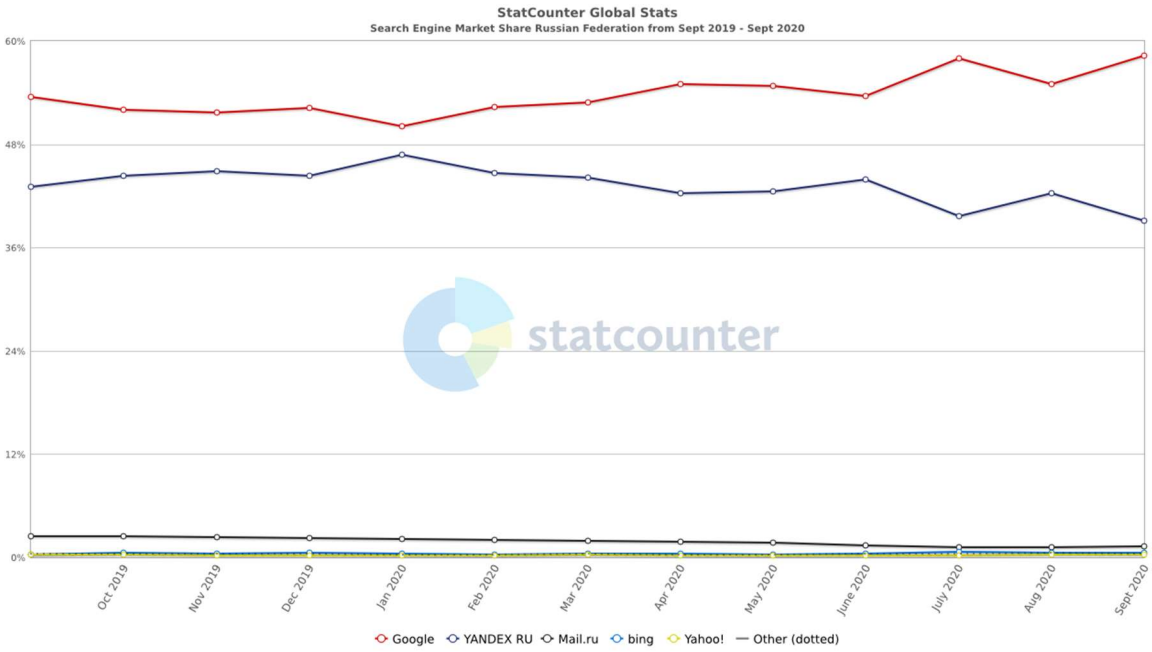


Figura 8 - Quota de mercado de motores de pesquisa na Federação Russa

Segundo a documentação da arquitetura do sistema de pesquisa de Yandex, o serviço recebe milhões de pedidos diariamente e, para que a resposta seja mais rápida, é utilizada a

indexação em conjunto com *clustering* de servidores e, em certos casos, até *clustering* de *clusters* de servidores. Todos os pedidos dos utilizadores são recolhidos pelo sistema "Metasearch", ou metapesquisa. O sistema analisa cada pedido relativamente à localização geográfica deste, a classe associada, entre outros parâmetros, fazendo de seguida um tratamento linguístico. Depois, o "metapesquisa" verifica se existem resultados recentes para a *query* do pedido. Os resultados para as pesquisas mais populares normalmente são armazenados por algum tempo na memoria do sistema para não haver necessidade de processar o pedido novamente, devolvendo-os instantaneamente [47].

Caso o resultado não se encontre na memória, o sistema "metapesquisa" redireciona o pedido para outro, designado por basic search servers" ou servidores de pesquisas básicas. O conjunto de servidores responsáveis pela pesquisa básica contém, citando a documentação, uma "cópia de Internet", em que, cada servidor processa uma parte especifica, permitindo distribuir a carga de forma mais eficiente.

Cada servidor da pesquisa básica devolve uma lista de documentos que contêm as palavras do pedido para o sistema metapesquisa, onde um algoritmo de *machine learning* designado por "MatrixNet" avalia os resultados e apresenta-os ao utilizador, como ilustrado pela Figura 9.

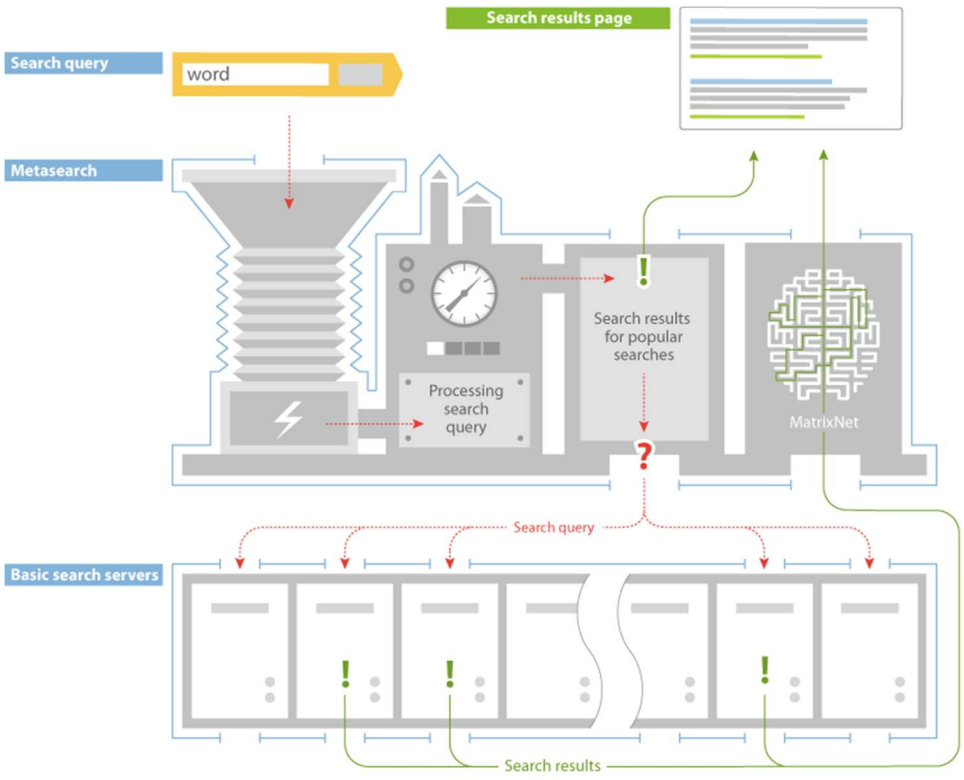


Figura 9 - Arquitetura de funcionamento de Yandex Search

O motor de pesquisa também permite parametrizar o pedido através de operadores [48], bem como ter em conta a morfologia da linguagem russa. O serviço possui também uma REST API com métodos GET e POST que devolvem respostas em formato XML, designado por "Yandex.XML", no entanto, a API possui um conjunto de medidas de segurança contra acessos automatizados. Para a utilização da API é necessário partilhar o endereço IP do utilizador e preencher o formulário CAPTCHA recolhido [49].

2.5.4. DuckDuckGO

O motor de pesquisa abreviado por DDG, foca-se em apresentar resultados relevantes respeitando a privacidade do utilizador. O que distingue DuckDuckGO dos restantes motores de pesquisa é a abordagem que exclui *profiling* dos utilizadores, ou seja, não armazena dados ou caraterísticas destes e apresenta os mesmos resultados para todos de acordo com o termo de pesquisa [50].

Os resultados são recolhidos a mais de 400 fontes de dados, obtidos através de *crowdsourcing* (sugestões e influências do público), ou através do *webclawler* próprio, designado por DuckDuckBot e armazenados nos índices de resposta. Para além do mencionado, o motor também possui endereços tradicionais nos resultados, obtidos através de múltiplos parceiros, sendo o mais comum o Bing [51].

Para além dos resultados, na parte superior da página, o DuckDuckGO apresenta resultados relevantes, designados por Instant Answers, recolhidos de várias APIs externas ou ficheiros de texto armazenados. Estes resultados também têm o nome de *zeroclickinfo,* isto é, informação em zero cliques. Como o nome indica, o propósito dos resultados é permitir ao utilizador visualizar a informação necessária sem ter de abrir hiperligações. No momento existem cerca de 1235 *Instant Answers* [52] [53].

Uma funcionalidade curiosa para aquisição de OSINT oferecida pela DuckDuckGO é a pesquisa através do site alvo, designada por "!Bangs". Isto é, quando é necessário pesquisar por informação num serviço específico, como por exemplo um utilizador de twitter, podem existir *bangs* que o permitam fazer, neste caso, seria "!twuser <utilizador>". O serviço no momento conta com cerca de 13 564 *bangs* [54], exemplificado na Figura 10.

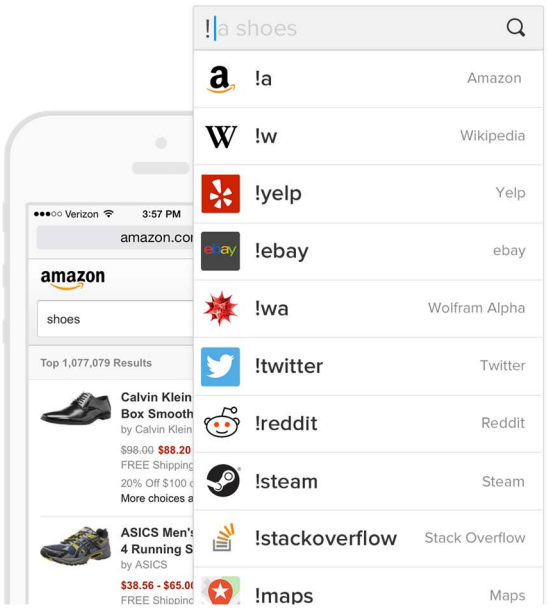


Figura 10 - !Bangs pela DuckDuckGO

Em relação às API, o serviço apenas dispõe da funcionalidade de *Instant Answers*. Para utilizar o motor de pesquisa através de aplicações externas será necessário fazer *scraping* de HTML.

2.5.5. Yahoo! Search

Um motor de pesquisa que não se destaca de qualquer forma dos referidos anteriormente, pelo contrário, utiliza o motor de Bing para as pesquisas [55]. Sendo um motor de pesquisa que faz uso de outro motor em estudo, não apresenta interesse na aquisição de OSINT, também pelo simples facto da sua API se encontrar desativada [56].

2.6. Redes sociais

Em relação as fontes de informação pública, uma rede social, definida no dicionário de Cambridge como “um website ou aplicação que permite às pessoas comunicar e partilhar informação na internet usando um computador ou telemóvel” [57], merece ser estudada e analisada quanto à facilidade de acesso e à quantidade de informação que é possível adquirir, respeitando os princípios de OSINT.

Um estudo realizado pela StatCounter, entre setembro de 2019 e setembro de 2020, representado pela Figura 11, mostra a dominância da Facebook, a nível mundial, com cerca de 74.58% da quota do mercado, seguido pelo Pintrest com 11.59%, Twitter com 6.88%, YouTube com 3.98%, Instagram com 1.58%, Tumblr com 0.54% e Reddit com 0.34% [58].

Em relação a Europa, a posição das redes sociais mantem-se, havendo apenas uma diferença ligeira de valores, enfatizando a dominância da Facebook com 81.77% [59].

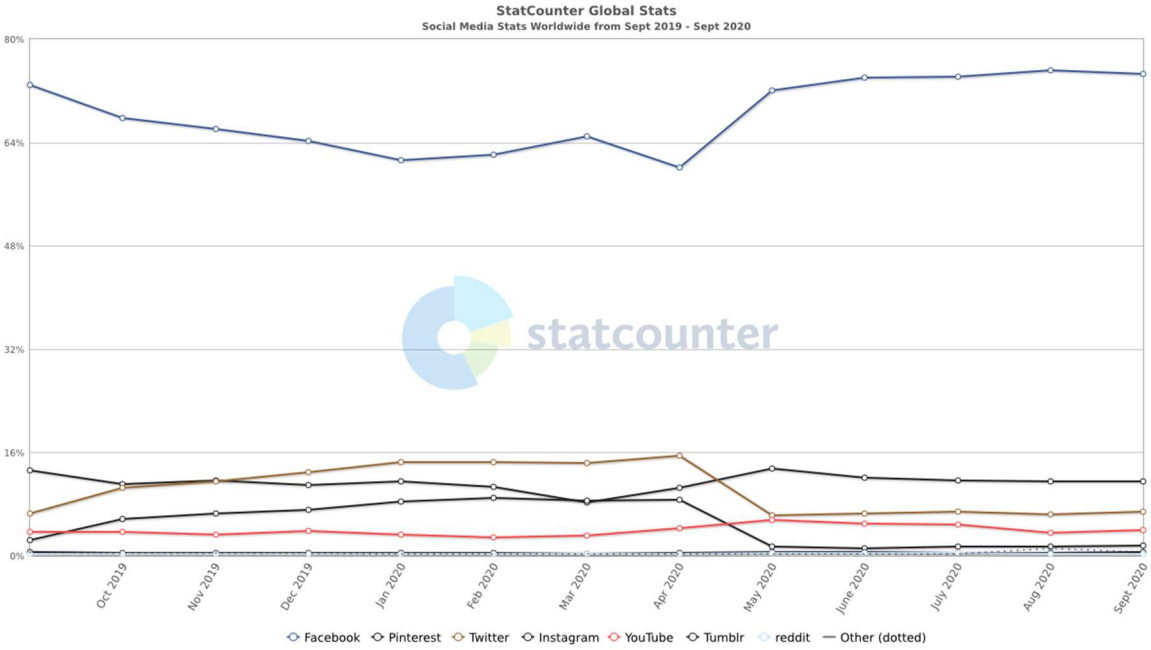


Figura 11 - Quota de mercado das redes sociais a nível mundial.

2.6.1. Facebook

A rede social mundialmente mais popular, com cerca de 2,6 biliões de utilizadores ativos mensalmente [60], permite aos seus utilizadores partilhar todo o tipo de informação pessoal, desde os dados básicos como: data e local de nascimento; género; sexualidade; número de telefone; entre outros, até a informação mais apurada como: amigos; a opinião em relação a vários temas através de grupos; publicações e comentários recentes; fotografias; entre outros.

A quantidade de informação que é possível extrair dos utilizadores registados torna o Facebook responsável pela privacidade destes e, ao mesmo tempo, uma fonte apetecível para muitas entidades. O utilizador tem o poder de decisão sobre a privacidade de vários elementos do seu perfil, podendo permitir ou não, acesso público a informação que partilha. Fugas de dados e acesso a informação privada numa rede social do tamanho de Facebook tem graves consequências tanto para a organização como para os seus utilizadores. O escândalo da privacidade de dados associado ao caso da Cambridge Analytica é um exemplo deste tipo de consequências [61].

A violação do RGPD e as suas consequências foram a possível causa da restrição de acesso ao GraphSearch, uma funcionalidade popular do Facebook utilizada pelos investigadores de OSINT [62].

Após uma maior análise das funcionalidades disponibilizadas pelo Facebook, verificamos a redução das capacidades de pesquisa, impossibilitando qualquer tipo de recolha de informação através das API fornecidas. A Graph API foi reduzida para possibilitar apenas a recolha das informações pessoais do utilizador da API, criação de publicações, grupos e eventos. O método "Search" [63] no momento é autorizado apenas para as aplicações do tipo "Workplace", como demonstrado pela Figura 12.

Figura 12 - Resultado da utilização do método `Search`

Apesar disto, existem formas de contornar algumas das limitações impostas pela Facebook. As limitações foram aplicadas sobretudo no ambiente de desenvolvimento. No que refere aos utilizadores normais, que iniciam a sua sessão através do navegador ou aplicação de telemóvel, não foram afetados pelas restrições e podem fazer qualquer tipo de pesquisa.

Para possibilitar pesquisas através do Facebook com as limitações atuais, existe um conjunto de soluções, como: IntelligenceX, Sowdust e SearchBook, que permitem, assumindo que o utilizador está com a sessão iniciada, construir URL a partir de um conjunto de parâmetros, que irá permitir usufruir das funcionalidades da rede social no contexto de utilizador normal [64].

2.6.2. Pinterest

O Pintrest é uma rede social americana com cerca de 416 milhões de utilizadores ativos mensalmente a nível mundial [65]. Permite aos seus utilizadores colar (*pin*) imagens nos seus quadros de avisos (*bulletin boards*), com o propósito de poder visualizá-las mais tarde.

Os utilizadores também podem utilizar colagens de outros e colar nos seus quadros, ou fazer uma colagem em conjunto.

Para os efeitos de aquisição de OSINT, uma conta de Pinterest pode obter as seguintes informações [66]:

- O nome da conta
- Nome de utilizador
- Foto de perfil
- Informação de perfil
- Seguidores
- Utilizadores seguidos
- Quadros de avisos
- Colagens

Figura 13 - Pesquisa por nome de utilizador no Pinterest

É também possível pesquisar por um utilizador de duas formas diferentes. Como o nome de utilizador no Pinterest é único, o *website* possibilita a sua pesquisa através do URL, como exemplificado pela Figura 13. No entanto, também é permitida a pesquisa através da API, desde que se possua uma chave de acesso, e a recolha de uma resposta em formato JSON, contendo os campos indicados, exemplo na Figura 14Figura 14. A vantagem na utilização da API está na possibilidade de pesquisa por nome de colagem ou quadro.

Figura 14 - Pesquisa por nome de utilizador na API de Pinterest

Caso seja necessário recolher a informação da página HTML no Pinterest terá de se fazer o seu respetivo *parsing*, que é dificultado pelo processo de geração aleatória de classes HTML através de funções de hash, com se pode observar na Figura 15Figura 15.

Figura 15 - Documento HTML de uma página no Pinterest

2.6.3. Twitter

O Twitter é uma rede social com cerca de 330 milhões de utilizadores ativos mensalmente [67] e é baseada em comunicações através de mensagens designadas por *tweets*. Os seus utilizadores podem publicar, gostar ou partilhar *tweets* dos outros, enquanto os não registados podem apenas visualizá-los [68].

O serviço REST API do Twitter oferece um conjunto enorme de operações no que se refere a pesquisa. É possível pesquisar por utilizador, ou grupo de utilizadores, especificando o(s) seu(s) id de utilizador ou nome de utilizador, recebendo em troca um user_object, com um conjunto enorme de informação acerca deste [69]. No entanto, as funcionalidades são disponíveis apenas aos utilizadores de desenvolvimento aprovados e que possuem um *token* de acesso [70].

Em caso de pesquisa com a sessão iniciada no browser, o Twitter permite pesquisar por *tweets* de forma avançada com um o número considerável de parâmetros, demonstrados pela Figura 16. A funcionalidade tem o nome de "Twitter Advanced Search" [71] [72].

Operator	Finds Tweets...
watching now	containing both “watching” and “now”. This is the default operator.
“happy hour”	containing the exact phrase “happy hour”.
love OR hate	containing either “love” or “hate” (or both).
beer -root	containing “beer” but not “root”.
#haiku	containing the hashtag “haiku”.
from:interior	sent from Twitter account “interior”.
list:NASA/astronauts-in-space-now	sent from a Twitter account in the NASA list astronauts-in-space-now
to:NASA	a Tweet authored in reply to Twitter account “NASA”.
@NASA	mentioning Twitter account “NASA”.
politics filter:safe	containing “politics” with Tweets marked as potentially sensitive removed.
puppy filter:media	containing “puppy” and an image or video.
puppy -filter:retweets	containing “puppy”, filtering out retweets
puppy filter:native_video	containing “puppy” and an uploaded video, Amplify video, Periscope, or Vine.
puppy filter:periscope	containing “puppy” and a Periscope video URL.
puppy filter:vine	containing “puppy” and a Vine.
puppy filter:images	containing “puppy” and links identified as photos, including third parties such as Instagram.
puppy filter:twimg	containing “puppy” and a pic.twitter.com link representing one or more photos.
hilarious filter:links	containing “hilarious” and linking to URL.
puppy url:amazon	containing “puppy” and a URL with the word “amazon” anywhere within it.
superhero since:2015-12-21	containing “superhero” and sent since date “2015-12-21” (year-month-day).
puppy until:2015-12-21	containing “puppy” and sent before the date “2015-12-21”.
movie -scary :)	containing “movie”, but not “scary”, and with a positive attitude.
flight :(	containing “flight” and with a negative attitude.
traffic ?	containing “traffic” and asking a question.

Figura 16 - Lista de operadores no Twitter Advanced Search

2.6.4. Instagram

O Instagram é uma rede social com cerca de bilhão de utilizadores ativos mensalmente [73], de partilha de fotos e vídeos, bem como de interação entre utilizadores através de gostos, comentários e mensagens [74].

No processo de aquisição de OSINT, para os utilizadores com sessão iniciada, o Instagram oferece as seguintes informações públicas:

- Caso o perfil alvo é privado;
    - Nome de utilizador;
    - Número de publicações, seguidores e seguidos;
    - Descrição;
    - Imagem de perfil;
- Caso o perfil alvo é público;
    - Publicações (fotos, vídeos);
    - Possível localização da publicação;
    - Comentários;

Desde 29 de junho de 2020 que o serviço Legacy API sofreu algumas alterações, migrando para Instagram Graph API, que partilha a mesma plataforma que o Facebook. Ou seja, operadores de pesquisa e acesso às informações públicas dos utilizadores estão restritos [75].

2.6.5. YouTube

O YouTube é uma plataforma de partilha de vídeos com cerca de 2 bilhões de utilizadores ativos mensalmente [76]. Contém caraterísticas de uma rede social, pois permite um conjunto de atividades relativas aos vídeos como: submeter, visualizar, avaliar, partilhar, adicionar a uma lista, reportar ou comentar.

Para o processo de aquisição de OSINT, a plataforma permite pesquisar por vídeos, canais ou listas de reprodução de acordo com critérios especificados aquando da utilização a sua API [77], no entanto é limitada a 100 resultados diários de utilização gratuita. Cada resultado devolvido tem o custo de 100, com o limite de 10 000 unidades de custo diário, ou seja, 100 resultados diários [78].

2.6.6. Tumblr

O Tumblr é uma rede social de *blogging* com cerca de 292.5 milhões de visitas mensais [79] e que permite aos seus utilizadores publicar “tumblelogs”, ou publicações curtas de *blogs*.

O que diferencia o Tumblr das restantes redes sociais é a sua estrutura livre e a enfase na customização das páginas pessoais [80].

Segundo a documentação da API do Tumblr, não existem métodos que permitam pesquisas por indivíduos ou publicações. No entanto, é possível pesquisar por publicações por *tags* como por exemplo `https://api.tumblr.com/v2/tagged?tag=gif`.

O serviço irá devolver um conjunto de publicações que contêm a *tag* em questão. Os pedidos a API estão limitados de seguinte forma:

- 300 chamadas à API por minuto, por cada endereço IP.
- 18,000 chamadas à API por hora, por cada endereço IP.
- 432,000 chamadas à API por dia, por cada endereço IP.

De resto, o serviço, assim como o Facebook, está de momento orientado à interação com apenas as informações do próprio utilizador e acesso ao seu feed.

2.7. Síntese

De forma resumida, o mundo de OSINT na Internet é bastante volátil. Existem muitas ferramentas de apoio na aquisição de informação de fontes abertas. Algumas especializam-se em fontes específicas, algumas agregam várias, algumas têm interação acessível e intuitiva, outras nem tanto. As soluções mais completas, geralmente, têm um custo associado e as gratuitas, por vezes, são prejudicadas pela dificuldade de utilização ou pela falta de flexibilidade.

Grande parte das ferramentas abordadas no documento acaba por se dedicar à auditoria de segurança das organizações e é, geralmente, utilizada pelos profissionais de cibersegurança. Consequentemente, a usabilidade destas aplicações acaba por ser prejudicada, pois a interface acaba por incluir linguagem técnica. Para além disto, muitas das ferramentas acabam por se especializar em informações específicas, isto é, recolher informação de apenas um serviço/fonte, e as que oferecem soluções mais abrangentes acabam por ter um custo monetário ou alto nível de complexidade, o que pode dificultar a sua utilização.

O tempo despendido na aprendizagem de cada uma das ferramentas poderá contestar com os benefícios da sua utilização por parte dos investigadores, bem como prejudicar uma investigação, caso a aplicação não ofereça os resultados pretendidos. As ferramentas de auditoria de segurança tradicionais, incluindo até as dedicadas ao OSINT, mesmo sendo úteis, não oferecem acesso à totalidade da informação sobre o indivíduo ou organização alvo [81], o que é completamente normal numa investigação. A necessidade de percorrer várias fontes, utilizar várias ferramentas e, possivelmente, aplicar uma análise específica a cada uma, é natural num processo de extração de conhecimento. No entanto, após revisão do processo de aquisição OSINT, podemos encontrar formas para a sua otimização.

Uma ferramenta que merece destaque é a Recon-ng, devido ao facto de se caraterizar como sendo uma ferramenta gratuita, de fonte aberta e com estrutura modular. A ferramenta oferece grande versatilidade nas pesquisas e replica uma interface familiar para os profissionais de cibersegurança, em forma da GUI na linha de comandos inspirada pelo Metasploit Framework. No entanto, a ferramenta acaba por ser limitada aos profissionais de cibersegurança pelo mesmo motivo. Um utilizador sem background informático terá dificuldade em usufruir das suas funcionalidades. A ferramenta tenta remediar isto com disponibilização dos resultados num serviço web local, permitindo visualizar e interagir come estes num formato mais familiar.

Para além disto, o acesso a informação tem ficado cada vez mais reduzido. Várias redes sociais fecharam "as portas" ao acesso automatizado da informação disponível publicamente, possivelmente para evitar potenciais desacordos com o RGPD. Serviços como a Graph API do Facebook limitaram acesso às funcionalidades cruciais para OSINT, tornando ferramentas que integravam o serviço irrelevantes.

De qualquer forma, o mundo tecnológico não para. Os regulamentos, as normas e as políticas estão em constante mudança. Enquanto a sociedade se adapta às evoluções digitais e tecnológicas, vão se criando vários serviços e meios de partilhar informação, tornando muitas ferramentas de aquisição de OSINT atuais desatualizadas, mas nunca obsoletas. Deste modo, os investigadores terão ao seu dispor, um número cada vez maior de ferramentas, muitas delas possivelmente desatualizadas e associadas a uma fonte.

O exemplo destas mudanças pode ser observado nas distribuições de máquinas virtuais, concebidas para agregar as ferramentas mais recentes e as mais populares. Enquanto este tipo de abordagem resolve o problema de agregação e seleção de ferramentas, acaba por ser uma solução a médio/curto prazo. Algumas destas distribuições, mesmo sendo relativamente recentes, deixaram de pertencer à lista de recomendações dos profissionais de OSINT, pela falta de atualizações.

Para além do mencionado, as ferramentas de OSINT, com algumas exceções, não conseguem chegar ao consenso na gestão da informação. Algumas ferramentas não oferecem qualquer tipo de gestão da informação e apenas devolvem resultados em dois ou três formatos predefinidos, que diferem de ferramenta para ferramenta. Isto obriga o investigador a gerir os resultados manualmente através de aplicações externas.

Por fim, os motores de pesquisa mostraram-se como ferramentas poderosas, através de mecanismos sofisticados de indexação e pesquisa na World Wide Web. As suas funcionalidades não foram ignoradas pela comunidade de inteligência, e são aproveitadas em várias aplicações, como por exemplo, a Recon-ng, com módulos de Google, Baidu, Bing ou Shodan.

3. FREE-OSINT

O seguinte capítulo descreve, a base das conclusões do capítulo anterior, os requisitos e caraterísticas para a solução desenvolvida. São definidos de forma detalhada, a metodologia seguida, as tecnologias envolvidas e o resultado do desenvolvimento.

3.1. Requisitos e caraterísticas

Para criar uma solução capaz de se adaptar ao ambiente com mudanças regulares é necessário separar os elementos persistentes dos mais voláteis em funcionalidades distintas, aplicando os últimos com a flexibilidade necessária. Será vantajoso também explorar algumas funcionalidades na REST API de um motor de pesquisa e verificar o que este tem para oferecer para além do que existe no serviço web.

Tendo em conta que a análise dos elementos voláteis e persistentes foi realizada previamente, em conjunto com a pesquisa exploratória de motores de pesquisa, temos informação que nos ajudará no processo da definição de requisitos.

O objetivo da aplicação desenvolvida será em tentar resolver os problemas mencionados a nível da aplicação, focando-se principalmente em três elementos, que são considerados como essenciais, que são: a usabilidade, estrutura flexível e adaptativa e a gestão de informação.

Com investimento na usabilidade, é permitido ao investigador focar no mais importante, que é a investigação e não, a aprendizagem das funcionalidades das diversas ferramentas. Uma aplicação fácil e intuitiva irá guiar o utilizador para o próximo passo lógico, melhorando assim a produtividade deste.

Uma estrutura flexível e adaptativa permitirá resistir a volatilidade, tanto nas fontes, como nas ferramentas do ambiente de OSINT. A informação e o formato dos dados deverão de ser tratados com a flexibilidade necessária, convertendo-se, caso necessário, num formato único.

A gestão da informação permitirá agregar os dados num ambiente, que possibilitará armazenar, interagir, alterar ou exportá-los noutro formato, sem a necessidade de recorrer a outras aplicações, facilitando assim a fase da recolha e análise da informação.

Para permitir a realização do processo de aquisição, seguindo o ciclo de OSINT exclusivamente através da aplicação, existe um conjunto de requisitos que esta terá de cumprir. Começando nos seguintes requisitos funcionais:

- Permitir a aquisição de dados de várias fontes;
- Permitir adquirir/extrair diferentes tipos de dados;
- Permitir a gestão de informação e o seu armazenamento;
- Permitir transformar ou processar os dados adquiridos;
- Permitir gerar relatórios;

Requisitos não funcionais:

- Partilhar a informação entre os passos do ciclo num formato persistente.
- Possuir de uma lógica de interação simples e intuitiva para qualquer tipo de utilizador.
- Permitir configurar vários elementos de apresentação e da gestão de informação.

Relativamente a lógica de interação, tentar-se-ão replicar vários mecanismos que podem ser encontrados no Maltego ou Spiderfoot, acrescentando também algumas funcionalidades específicas.

- *Drag & Drop* dos objetos de dados;
- Duplo clique no objeto para editar os dados;
- Interligar os objetos com um clique do rato;
- Seleção de múltiplos objetos de dados simultaneamente;
- *Zoom in/Zoom out*;
- Atalhos para as funcionalidades;
- Sincronizar os dados do grafo com a área de trabalho;
- Diferenciar os dados utilizando cores diferentes de representação;
- Criar os dados a partir do ambiente de interação;

Funcionalidades adicionais específicas:

- Compor um termo de consulta utilizando os dados do ambiente de interação;
- Abrir a página web no navegador a partir do ambiente de interação;
- Manipulação do tamanho dos objetos;

A partir dos requisitos referidos cria-se um conceito de aplicação com as seguintes caraterísticas:

- Uma estrutura modular que separa os processos de aquisição, processamento e documentação em funcionalidades distintas (ex Pesquisa, Processamento e Documentação). Permitirá cumprir os requisitos de aquisição de várias fontes, adquirir/extrair diferentes tipos de dados e construir relatórios.
- Definição de um formato único para os dados recolhidos. Permitirá transformar ou processar estes e partilhá-los entre os passos do ciclo OSINT.
- Criação de um ambiente de interação com a informação recolhida/transformada. Permitirá criar uma lógica de interação simples e intuitiva para qualquer tipo de utilizador, bem como suplementar um meio para transformar ou processar os dados manualmente.
- Criação de um sistema de armazenamento de dados, permitirá cumprir o requisito de gestão e armazenamento da informação e possíveis configurações.

Com a estrutura modular, deverão de existir módulos necessários para que seja possível, através da aplicação, completar pelo menos um ciclo de aquisição OSINT.

Por fim, a solução deverá de usufruir do serviço REST API do motor de pesquisa da Google.

3.2.Metodologia de desenvolvimento da aplicação

Após o estudo do contexto atual e definição do problema, para o protótipo e as funcionalidades base do projeto, o desenvolvimento realizado seguiu a metodologia em cascata, com objetivos/funcionalidades bem definidas, em conjunto com sprints semanais. As últimas etapas do projeto foram desenvolvidas através de metodologias ágeis, com testes de usabilidade e melhorias na interação.

Para usufruir dos elementos persistentes e respeitar a volatilidade dos restantes é beneficial seguir o exemplo da Maltego e Recon-ng, criando uma arquitetura modular e definindo um formato abstrato para os dados recolhidos.

3.3. Tecnologias e ferramentas

Neste capítulo são descritas as tecnologias que são o fundamento da solução proposta.

3.3.1. Linguagem e *framework*

A aplicação é codificada utilizando a *framework* .NET 4.7.2, a base de linguagem C#, devido à sua enorme flexibilidade.

As aplicações desenvolvidas com a *framework* .NET podem fazer *deploy* num vasto conjunto de sistemas operativos ou plataformas cloud. Para que seja possível executar aplicações de Windows Forms no Linux basta instalar o "Mono" [82], plataforma de aplicações destinada a funcionamento *cross-platform*. A linguagem é compilada e executada rapidamente, com um conjunto vasto de tipos de dados e bibliotecas, sem necessitar de ficheiros de cabeçalho e é orientada a objetos.

Sendo uma linguagem fortemente apoiada pela Microsoft também usufrui de várias bibliotecas de apoio da *framework* .NET que permitem construções de janelas e elementos visuais de forma rápida e acessível, permitindo ao programador focar na funcionalidade e não na representação.

A interoperabilidade aumenta o potencial de uma aplicação com arquitetura modular, o que é suportado pelo C#. Isto permitirá o desenvolvimento de funcionalidades que usufruíram de código das outras linguagens de programação. Para além disto, C# é também uma linguagem orientada a componentes, através dos conceitos de métodos, propriedades, eventos e atributos, permitindo criar componentes auto-descritivos, designados por *assemblies* [83].

3.3.2. Formato dos dados

Para a definição do formato único podemos nos inspirar pela ferramenta Maltego em que, cada nó é construído e representado com *string*, o que limita o formato dos dados para apenas texto. No entanto, módulos dedicados a pesquisa web podem devolver resultados em forma de endereço da página que contém a informação. Desta forma, os dados estão encapsulados no formato de página web, oferecendo acesso ao tipo de informação que esta pode conter,

como texto, fotos, vídeos e muito mais. Claramente, o acesso depende da disponibilidade da página.

Neste caso, o formato único dos dados tomará a forma de um nó na arvore e não no grafo. Esta decisão deve-se a dois fatores. Para o Maltego, em muitos casos, as informações são apresentadas em forma de grafo sem justificação, podendo ser facilmente transformados numa arvore, com nós que têm apenas um parente. Em segundo lugar, para além de poder apresentar as informações num ambiente gráfico, também se pretende disponibilizar um acesso rápido para os dados em forma de uma lista expansível pois, em situações de grandes quantidades de informação torna-se difícil encontrar um elemento específico.

Tendo em conta que *framework* .NET possui já um componente para a interação e apresentação de entidades em forma de arvore, será interessante a escolha deste para o uso na aplicação, para reduzir o tempo de desenvolvimento e testes. Por este motivo, a classe escolhida para servir de formato único na aplicação tem a designação TreeNode, e é suportada por um conjunto vasto de tecnologias:

- .NET 5.0;
- .NET Core 3.0, 3.1;
- .NET Framework 1.1, 2.0, 3.0, 3.5, 4.0, 4.5, 4.5.1, 4.5.2, 4.6, 4.6.1, 4.6.2, 4.7, 4.7.1, 4.7.2, 4.8;

A TreeNode é uma classe pertencente à TreeView, uma componente da *framework* .NET responsável pela representação visual de arvores de nós, para demonstrar os dados de forma hierárquica. A classe implementa as interfaces IClonable e ISerializable, o que facilita a transformação da informação para o formato único.

3.4. Implementação

Neste capítulo são apresentados os elementos chave da solução desenvolvida divididos em subcapítulos. São demonstrados os mecanismos e as estruturas que permitem aplicar conceitos de modularidade e gestão da informação. São também apresentadas as decisões tomadas e as funcionalidades implementadas que corresponde ao objetivo de usabilidade.

3.4.1. FREE_OSINT_LIB

O primeiro elemento da solução desenvolvida e que permite implementar uma estrutura modular é a biblioteca FREE_OSINT_LIB. Esta biblioteca é o ponto de ligação entre os

módulos e a aplicação principal, que se deve à partilha de classes e interfaces com as regras que um módulo deverá de cumprir.

Cada modulo na *framework* .NET é uma aplicação com *assembly* própria. Para a partilha das funcionalidades, as aplicações com *assemblies* diferentes necessitam de uma referência comum. De modo a ser reconhecido pela aplicação, o modulo terá de implementar a interface IGeneral_module, utilizada para a sua identificação. Para as restantes funcionalidades terá que de se implementar a interface correspondente da biblioteca, o diagrama de classes referente à biblioteca pode ser observado na Figura 17. Neste momento, os módulos estão categorizados em: pesquisa, processamento e documentação, com possíveis caraterísticas em forma de configurável e interativo.

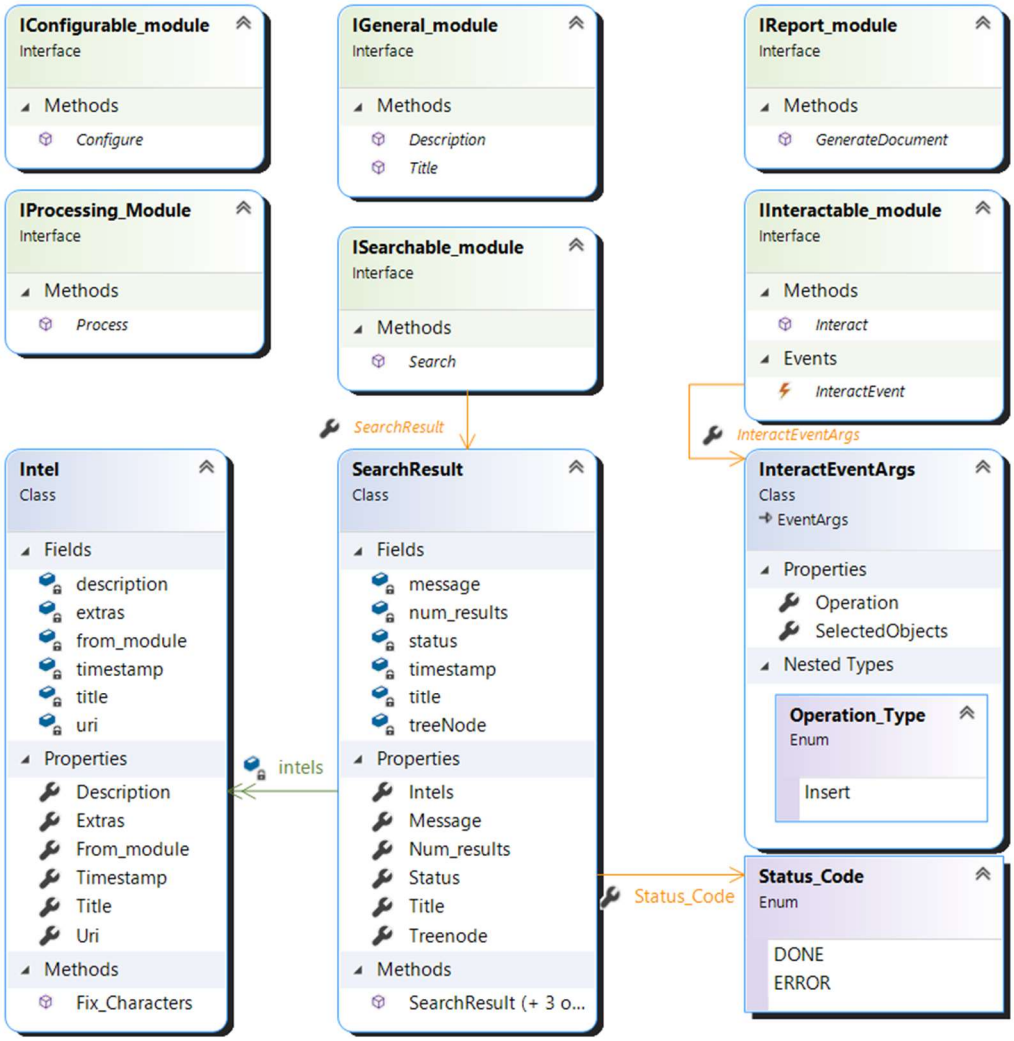


Figura 17 – Diagrama de classes da biblioteca FREE_OSINT_Lib

Para melhorar a interação dos módulos com a aplicação principal, a biblioteca permite lançar eventos através da classe InteractEventArgs, indicando o tipo de operação e, caso hajam, objetos para inserir ou abrir no navegador embutido.

O código responsável pelo carregamento dos módulos para a memória, ilustrado pela Figura 18, cria um conjunto de restrições. A primeira linha do código recolhe todas as subdiretorias da pasta *modules* localizada na raiz da aplicação. De seguida, para cada diretoria verifica a existência de uma assembly no ficheiro ".exe". É importante ter em conta que a aplicação assume que o nome do ficheiro é idêntico à diretoria onde se encontra, isto é, um módulo com o ficheiro de assembly "exemple_module.exe" terá de se localizar na diretoria designada por "exemple_module", contida dentro da *modules*.

A última restrição é a necessidade do modulo implementar sempre a interface IGeneral_module, pois esta é usada no carregamento para identificar o módulo. Caso o módulo desenvolvido seja do tipo DLL, basta inseri-lo na pasta *libs* sem restrições, exceto as de implementação de interfaces.

```
string[] module_directories = System.IO.Directory.GetDirectories(General_Config.modules_directory);
                for (int i = 0; i < module_directories.Length; i++)
                {
                    try
                    {
                        var moduleAssembly = System.Reflection.Assembly.LoadFrom(module_directories[i] +
                         "/" + module_directories[i].Split('\\')[1] + ".exe");
                        var moduleTypes = moduleAssembly.GetTypes().Where(t =>
                           t.GetInterfaces().Contains(typeof(IGeneral_module)));
                        modules = moduleTypes.Select(type =>
                        {
                            loadingForm.txtCurrentModule.Invoke((MethodInvoker)delegate
                            {
                                loadingForm.txtCurrentModule.Text = " Loading: " + type.FullName;
                            });
                            return (IGeneral_module)Activator.CreateInstance(type);
                        });
                        module_list.AddRange(this.modules.ToList());
                    }
                    catch (Exception e)
                    {
                        MessageBox.Show(e.Message);
                    }
                }

                string[] fileEntries = Directory.GetFiles(General_Config.modules_directory + "/" +
General_Config.lib_directory);
                for (int i = 0; i < fileEntries.Length; i++)
                {
                    try
                    {
                        var moduleAssembly = System.Reflection.Assembly.LoadFrom(fileEntries[i]);
                        var moduleTypes = moduleAssembly.GetTypes().Where(t =>
                           t.GetInterfaces().Contains(typeof(IGeneral_module)));
                        modules = moduleTypes.Select(type =>
                        {
                            loadingForm.txtCurrentModule.Invoke((MethodInvoker)delegate
                            {
                                loadingForm.txtCurrentModule.Text = " Loading: " + type.FullName;
                            });
                            return (IGeneral_module)Activator.CreateInstance(type);
                        });
                        module_list.AddRange(this.modules.ToList());
                    }
                    catch (Exception e)
                    {
                        MessageBox.Show(e.Message);
                    }
                }
```

Figura 18 - Código para o carregamento dos módulos

As interfaces criadas refletem a abstração e flexibilidade da comunicação entre os módulos e a aplicação. As interfaces definidas são descritas de seguinte forma:

- Interfaces que identificam um módulo:
  - IGeneral_module – utilizada para o carregamento do módulo na fase inicial da aplicação e contém os seguintes métodos que o descrevem:
    - Title – título do módulo.
    - Description – descrição do módulo.
- Interfaces que identificam uma funcionalidade especifica:
  - ISearchable_module – carateriza o módulo com capacidade de realização de pesquisas. Tem apenas o seguinte método:
    - Search – realiza a pesquisa recolhendo a query em forma de string e possíveis parâmetros opcionais em forma de lista de objetos. O método devolve a resposta em forma de SearchResult.
  - IProcessing_module – carateriza o módulo com capacidade de processamento de TreeNodes. Tem apenas o seguinte método:
    - Process – recolhe um TreeNode para extração de informação e devolve um TreeNode com a informação processada.
  - IReport_module – carateriza o módulo com capacidade de construção de relatórios através do seguinte método:
    - GenerateDocument – constrói um relatório a partir de uma lista de TreeNodes.
- Interfaces que identificam uma funcionalidade acrescida:
  - IConfigurable_module – caracteriza o modulo com capacidade de configuração. Tem apenas o seguinte método:
    - Configure – invoca um algoritmo de configuração do modulo.
  - IInteractable_module – caracteriza o módulo com capacidade de interação por parte do utilizador. Possui um evento para comunicar à aplicação a necessidade de inserção de novos dados. Tem apenas o seguinte método:
    - Interact – recebe uma *query* para permitir pesquisas recursivas e invoca um ambiente de interação com o módulo.

3.4.2. Aplicação principal (FREE-OSINT)

A aplicação principal é implementada como o elemento central que faz a leitura dos módulos e os permite integrar no processo de aquisição. O início da aplicação é caraterizado pelo fluxo de carregamento dos módulos, demonstrado pela Figura 19. São carregados os módulos válidos na memória e é aberta a janela principal. Todos os módulos são opcionais e não afetam a correta execução da aplicação e utilização das suas funcionalidades base. É possível utilizar a aplicação sem um único módulo presente, no entanto, sem os módulos de pesquisa e documentação, não terá nenhuma forma de adquirir dados e extrair a informação criada para um formato desejado.

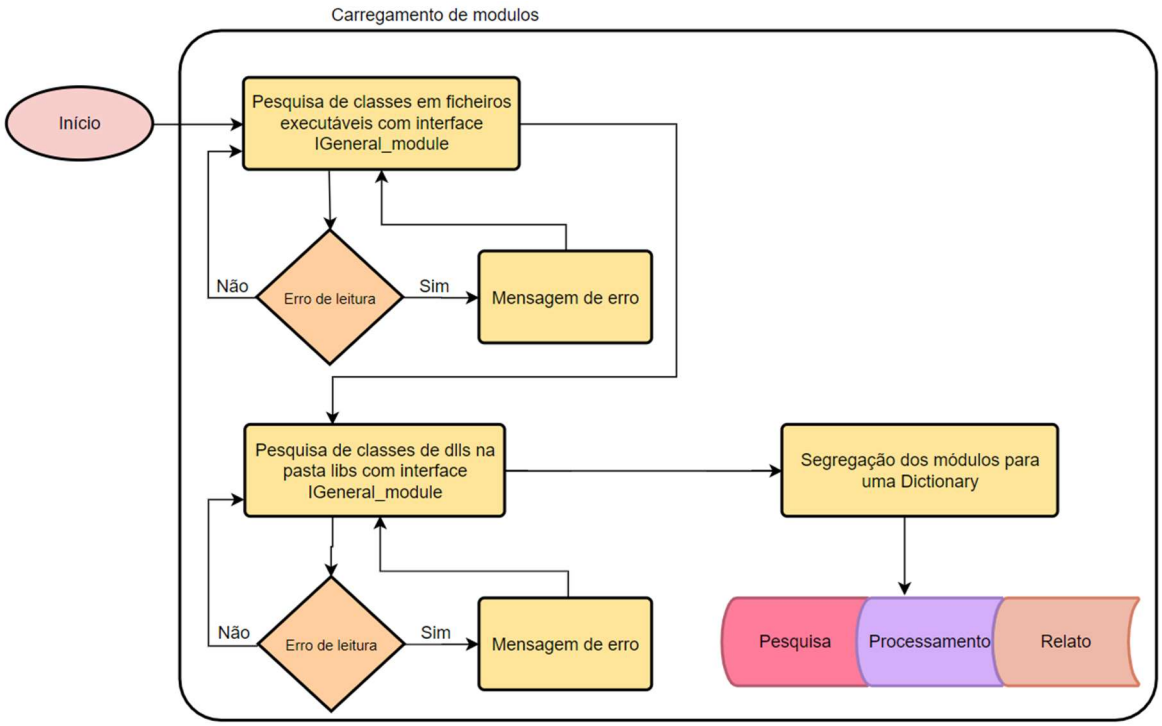


Figura 19 - Fluxo de carregamento dos módulos

Antes de aceder à janela principal, é nos apresentada a janela de início, que se pode observar na Figura 20, permitindo de forma rápida aceder aos ambientes de trabalho anteriores ou criar um de raiz.

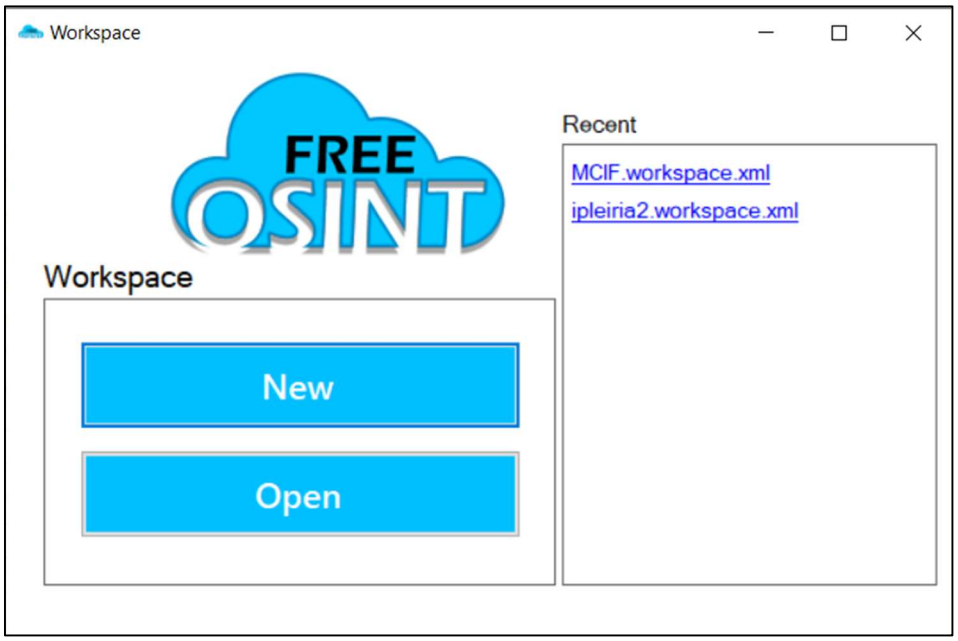


Figura 20 - Janela de início

A janela principal, demonstrada na Figura 21, pode ser considerada como a área principal de trabalho. É onde os dados são apresentados e alterados. Esta tem o nome do *workspace* no contexto da aplicação. É permitido criar, abrir ou guardar um *workspace*, através da opção *File*, no canto superior esquerdo.

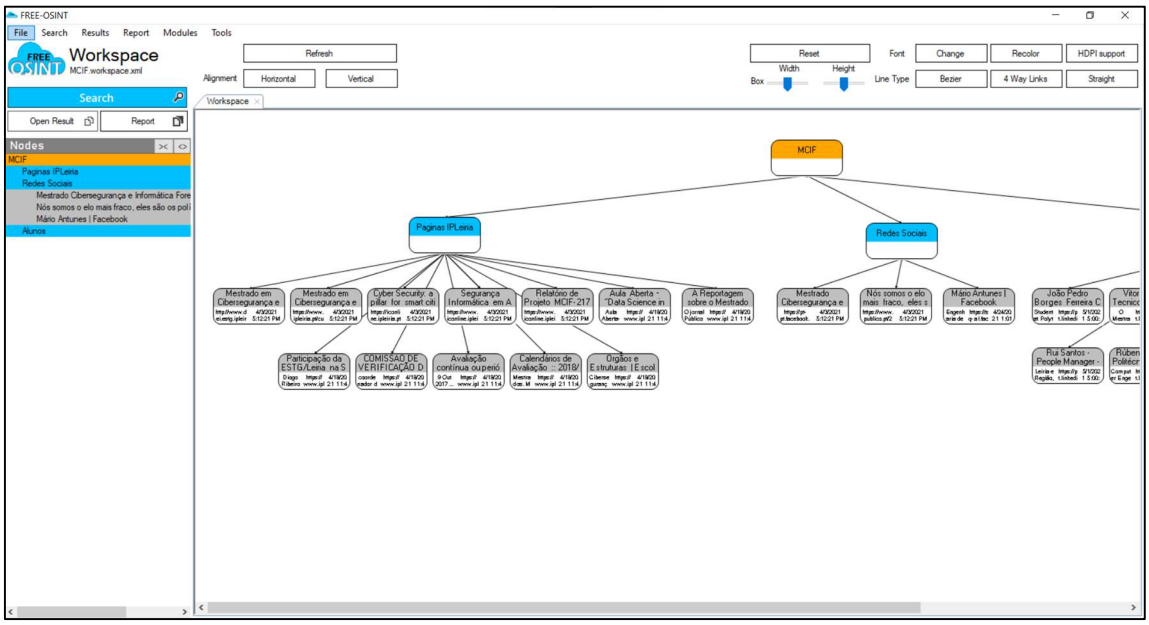


Figura 21 -Janela principal

O *workspace* é representado por duas componentes, uma TreeView do lado esquerdo e uma NodeDiagram no centro. Ambas partilham os dados e o objetivo de apresentar a informação ao utilizador, mas também permitem manipulá-la e qualquer alteração realizada numa

componente irá afetar a outra. Em termos de codificação, o *workspace* define um projeto de investigação, tendo uma lista de alvos, ou *target* no contexto da aplicação, e dados destes. Ambos, *workspace* e *target*, têm uma designação única utilizada como identificação.

O *workspace*, para além da lista de alvos e título, possui também uma vista em árvore sincronizada com a lista de alvos utilizada para manipulações no menu principal. Para além disto o *workspace* armazena também informações da posição, em forma de coordenadas XY, e cor dos dados no formato ARGB, valor inteiro de estrutura de 32-bits, exemplificado pelo ficheiro XML na Figura 22. A informação é armazenada num ficheiro com sufixo de "workspace.xml" para ser distinguidos dos restantes ficheiros XML e ajudar na procura do relevante.

```
▼<Workplace>
 ▼<Target value="MCIF" x="776" y="40" color="-23296">
  ▼<Node value="Paginas IPLeiria" x="288" y="144" color="-16728065">
   ▼<Node value="Mestrado em Cibersegurança e Informática Forense MCIF" x="24" y="240" color="-4144960">
      <Node value="http://www.dei.estg.ipleiria.pt/mcif/"/>
      <Node value="4/3/2021 5:12:21 PM"/>
    </Node>
   ▶<Node value="Mestrado em Cibersegurança e Informática Forense" x="128" y="240" color="-4144960">
    ...
    </Node>
   ▶<Node value="Cyber Security: a pillar for smart cities - IC-Online" x="232" y="240" color="-4144960">
    ...
    </Node>
   ▶<Node value="Segurança Informática em Ambientes de Avaliação Escolar - IC-Online" x="336" y="240" color="-4144960">
    ...
    </Node>
   ▼<Node value="Relatório de Projeto MCIF-2170099-Nuno Anacleto.pdf" x="440" y="240" color="-4144960">
```

Figura 22 - Ficheiro XML da *workspace*

A classe General_Config contém as configurações base da aplicação como a diretoria para os módulos e bibliotecas, ou *libs* no contexto da aplicação, a hierarquia de cores para os dados, e a diretoria para o armazenamento de resultados e *workspaces*.

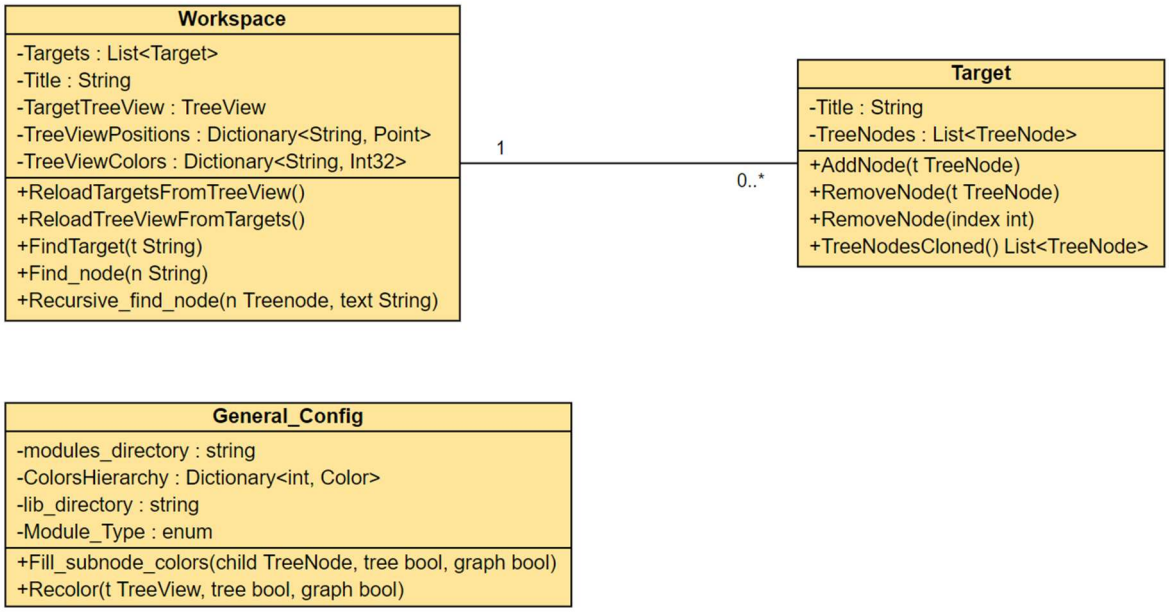


Figura 23 - Diagrama de classes (Aplicação Principal)

A componente NodeDiagram é proveniente da biblioteca desenvolvida pelo utilizador de GitHub, drake7707, designada por NodeControl [84], e que foi modificada para os efeitos desta aplicação sob licença de MIT. Esta componente é utilizada para mostrar a informação em forma de caixas com informação interligadas. É possível criar, editar, associar e eliminar os dados a partir da NodeDiagram. Para criar um nó basta carregar com botão direito do rato num espaço vazio. A aplicação irá permitir criar dois tipos de nó: *Target* e *Intel*. A única distinção entre estes é a hierarquia do nó criado. Ao criar um nó *Target*, está a ser criado nó raiz, ou seja, um nó sem parente e para lhe posteriormente associar nós. Ao criar um nó *Intel*, estamos a criar um nó que por omissão é um filho e é automaticamente atribuído para o nó *Unassigned*.

O *layout* é inspirado, em parte, pelo Adobe Photoshop, em que para além da visualização gráfica de componentes, existe um painel com estes em forma de lista [85]. O painel acima do NodeDiagram tem um conjunto de opções ligadas a representação gráfica. As duas "TrackBars" com nomes *width* e *height* são responsáveis pelo tamanho das caixas desenhadas e a opção "Reset", acima destes, permite voltar para o tamanho predefinido. As opções com a descrição "Alignment", "Horizontal" e "Vertical" permitem organizar automaticamente as caixas de informação na horizontal e na vertical. Caso o utilizador pretende mudar o estilo do texto da representação gráfica, a opção "Font Change" irá invocar uma janela de seleção de fontes instaladas na máquina e consequentemente alterar o estilo do texto nas caixas. As opções descritas pelo "Line Type", como: "Bezier", "4 Way Links"

e “Straight”, modificam o estilo dos ponteiros de ligação para um destes estilos. A última opção é destinada para computadores com HDPI, designada por “HDPI support”, para permitir melhorar a qualidade de imagem nestes.

O acesso aos módulos de pesquisa encontra-se na opção *Search ->Modules,* ou na opção “Search”. Esta opção irá abrir a janela de seleção de módulos de pesquisa, representado na Figura 24. Esta janela contém a lista de módulos de pesquisa e que implementam a interface ISearchable_module. Ao selecionar um módulo poderá ter acesso às opções de interação ou configuração deste. As opções apenas estarão disponíveis para módulos que implementam a respetiva funcionalidade. Após selecionar a opção “Interact”, a aplicação irá executar o método “Interact()” do modulo, pois este implementa a respetiva interface, ou seja, IInteractable_module. A seleção da opção “Configure” tem um comportamento semelhante, executando o método “Configure()” da interface IConfigurable_module. O bloco de texto com o cabeçalho “Description” é preenchido pelo método “Description()” da interface IGeneral_module. A opção “Search using X modules”, em que o X indica o número de módulos, abre a janela de pesquisa, ilustrada na Figura 25, com os módulos selecionados.

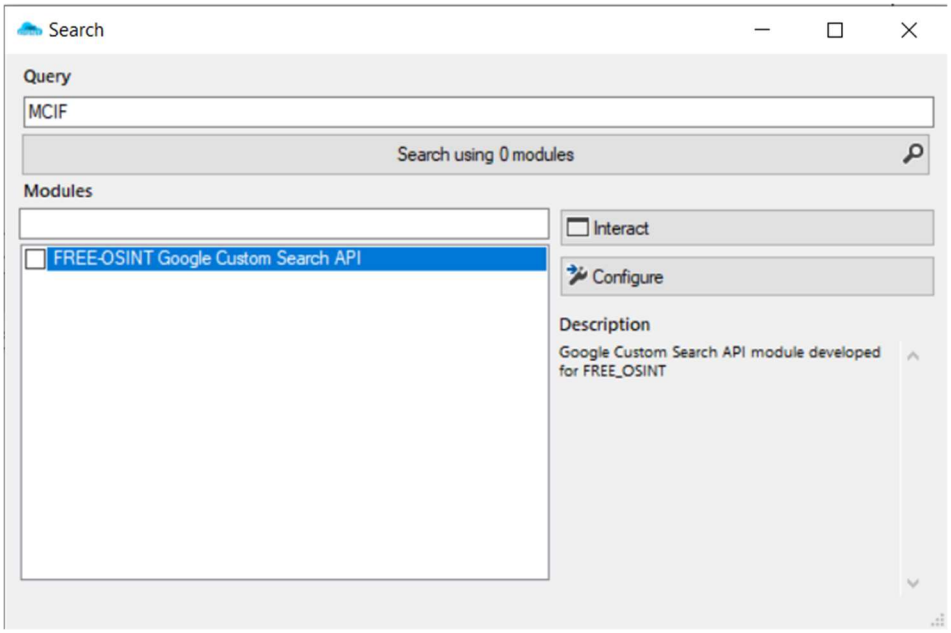


Figura 24 - Janela de seleção de módulos de pesquisa

A janela de pesquisa começa por listar os módulos e criar uma *Thread* onde é executado o método “Search()” para cada módulo. O estado do decorrer e fim da execução são representados no cabeçalho “Status”, em conjunto com a quantidade de resultados obtidos no cabeçalho “Results”. As mensagens relativas ao fim de execução de cada módulo são

apresentadas na caixa de “Events”. Ao fim da execução da pesquisa por todos os módulos, a opção “Done” estará disponível. Após selecionar a opção, abre-se a janela de resultados como um separador novo da aplicação principal, ilustrado na Figura 27.

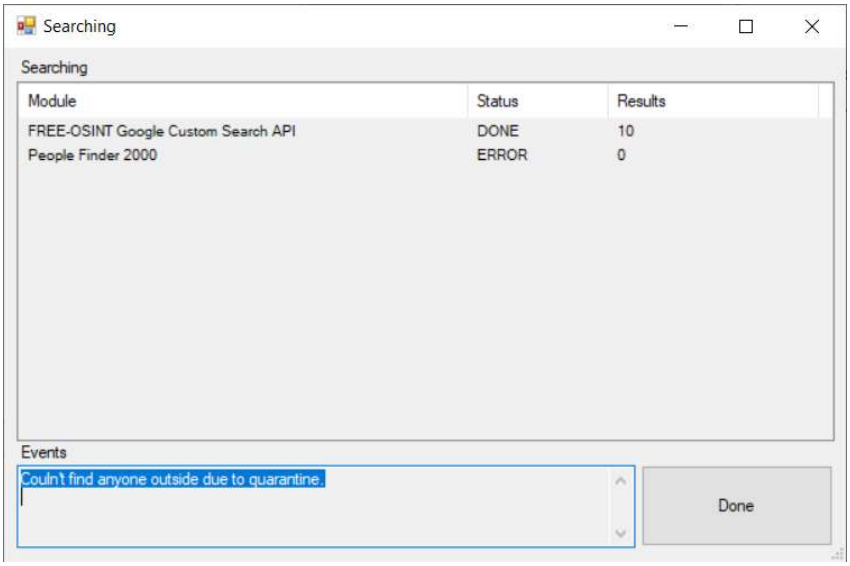


Figura 25 - Janela de pesquisa

Esta janela contém os resultados da pesquisa de todos os módulos em forma de TreeView. No centro da janela temos acesso ao navegador web da Cromium Embedded Framework (CEF) do Marshall A. Greenblatt [86]. O navegador permite aceder aos endereços de resultados obtidos, bastando selecionar o nó que contém um endereço web para que o navegador o abra. Caso existe algum problema de visualização da página, no canto superior esquerdo existe a opção “Browser” que irá abrir o endereço atual no navegador predefinido do sistema operativo.

Os resultados obtidos dos módulos não são automaticamente adicionados ao *workspace*. O objetivo da janela de resultados é permitir ao utilizador selecionar os dados relevantes manualmente. É possível armazenar os resultados num ficheiro com sufixo “result.xml” para o distinguir dos restantes ficheiros, exemplificado pela Figura 26.

```
▼<Node value="FREE-OSINT Google Custom Search API">
  ▼<Node value="Polytechnic of Leiria">
     <Node value="IPLeiria develops its R&D activities through 13 research units
     <Node value="https://www.ipleiria.pt/home/"/>
     <Node value="5/2/2021 1:56:30 PM"/>
   </Node>
  ▶<Node value="IPLeiria - Home | Facebook">
   ...
   </Node>
  ▶<Node value="Politécnico de Leiria">
   ...
   </Node>
  ▶<Node value="O seu browser não é suportado.">
```

Figura 26 - Ficheiro XML de resultados

Para além disto, é possível interagir com o TreeView de resultados através do botão direito do rato, ilustrado pela Figura 28. O seguinte conjunto de opções será apresentado:

- Insert – inserir o nó selecionado no workspace, num dos seguintes objetos:
  - "Nome do target existente" – adicionará o nó para o alvo existente.
  - New Target – permite criar um alvo e adicionar lhe o nó.
  - Unassigned – adicionará o nó para o alvo predefinido.
- Open URL – irá abrir o endereço do nó selecionado, ou o primeiro endereço dos filhos, através de uma das seguintes formas:
  - Chromium – navegador da aplicação.
  - Predefined Browser – navegador predefinido do Windows.
- Process Module – abre a janela com lista de módulos de processamento e aplica o modulo ao nó selecionado.
- Remove – remove o nó da lista de resultados.

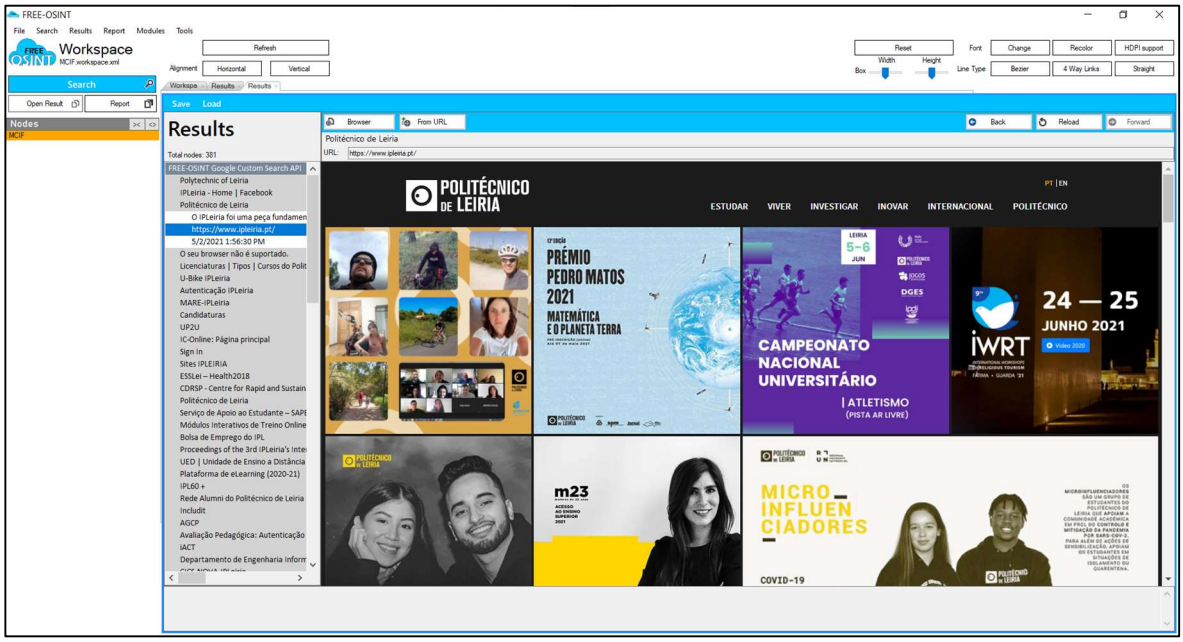


Figura 27 - Separador de resultados

No painel superior da aplicação também temos acesso a um conjunto de opções. A opção browser irá enviar o endereço corrente para o navegador predefinido. A opção "From URL" irá abrir uma janela para adicionar o endereço atual para o *workspace*. No canto superior direito encontram-se opções do navegador como: voltar atrás, recarregar e voltar a frente.

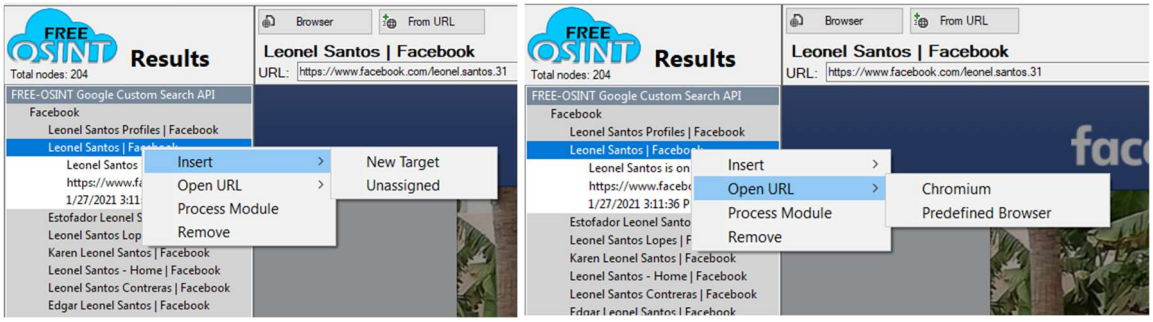


Figura 28 - Interação com nós de resultados

Para aceder aos módulos de relato, no topo da janela principal encontra-se a opção *Report,* ilustrado pela Figura 29. Ao selecionar "Modules" será invocada a janela de seleção de módulos de construção de relatórios, ilustrado pela Figura 30.

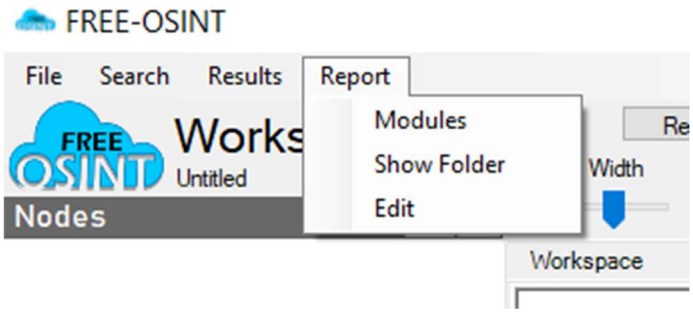


Figura 29 - Opção Report da janela principal.

Na janela de seleção de módulos de construção de relatórios são apenas listados módulos que implementam a interface IReport_module. Após seleção do módulo pretendido e carregar na opção OK será invocada a janela de seleção de nós, ilustrado pela Figura 31. Com os nós selecionados, o módulo a selecionar deverá poder construir um relatório.

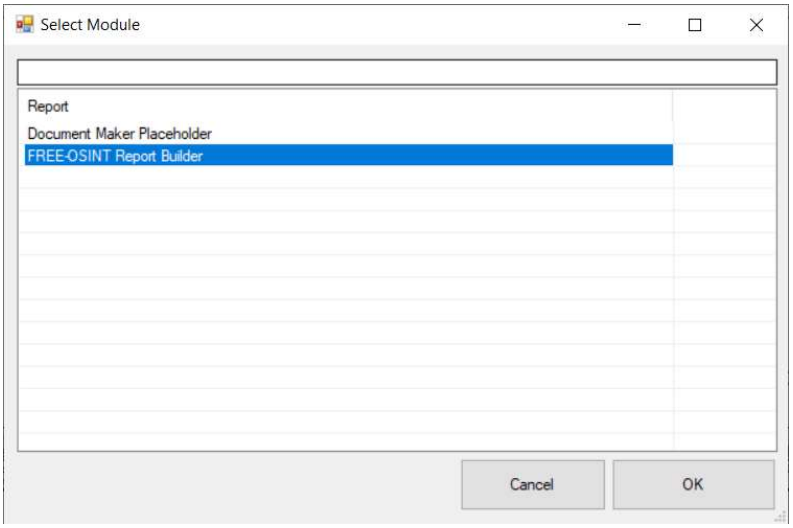


Figura 30 - Janela de seleção de módulos de construção de relatórios

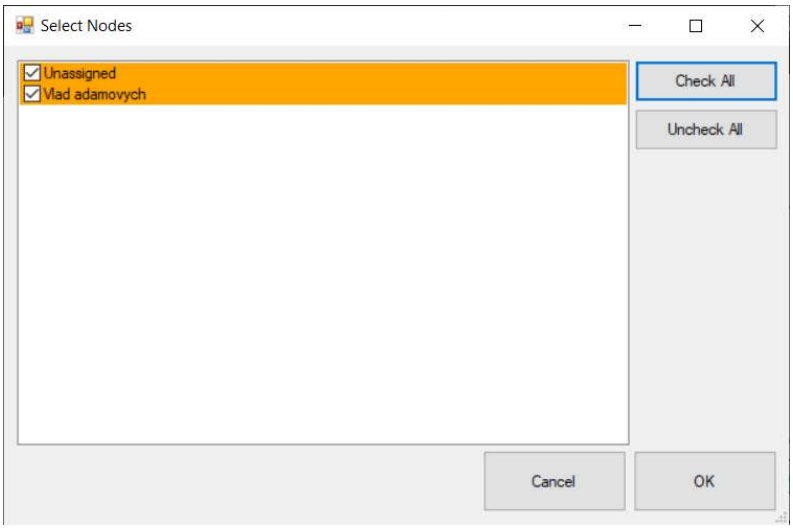


Figura 31 - Janela de seleção de nós para construção de relatórios

A última opção designada por "Options" na secção "Tools" permite aceder à janela de configurações. Esta contém opções como: lista de cores de contentores que são atribuídos pela ordem de hierarquia; número de contentores por linha para organizações automáticas; bem como a opção do modo de performance, que permite reduzir a complexidade no desenho do *workspace*, melhorando assim o desempenho.

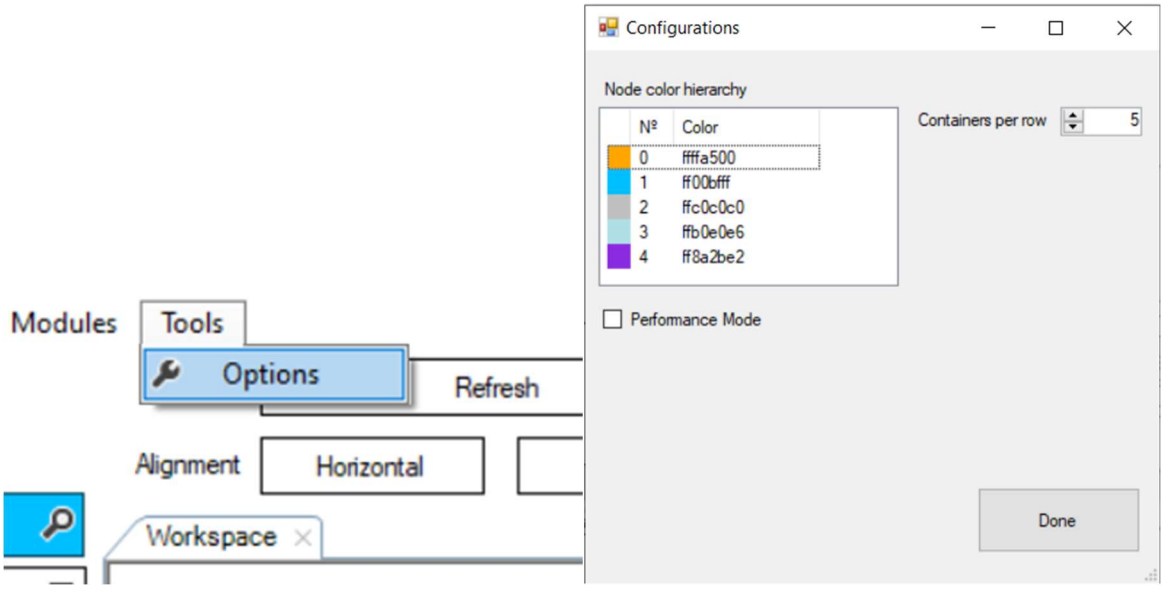


Figura 32 - Acesso a configuração

Todas as alterações feitas nas configurações são armazenadas no ficheiro designado por "config.xml". Este, para além das configurações, também contém a lista de *workspaces* utilizados recentemente, para facilitar a retoma da investigação na próxima interação.

```
<config>
  <colors>
    <first>-23296</first>
    <second>-16728065</second>
    <third>-4144960</third>
    <fourth>-5185306</fourth>
    <fifth>-7722014</fifth>
  </colors>
  <recent>
    <project>C:\Users\Nekki\Documents\FREE-OSINT\MCIF.workspace.xml</project>
    <project>C:\Users\Nekki\Documents\FREE-OSINT\ipleiria\ipleiria2.workspace.xml</project>
  </recent>
</config>
```

Figura 33 - Ficheiro de configuração

3.4.3. CefSharp

Nas primeiras fases de desenvolvimento, a aplicação fazia uso do navegador da *framework* .NET embutido para visualizar os resultados de pesquisas. Muito rapidamente se observou a simplicidade e a limitação deste navegador, com erros de execução de JavaScript e modo de visualização móvel. Concluiu-se que a aplicação teria de permitir a abertura dos endereços web num browser externo, mas isto irá impactar a experiência e a facilidade de interação do utilizador.

Decidiu-se então que a aplicação deveria permitir a abertura das hiperligações tanto num navegador externo como no navegador embutido, com a condição que o navegador embutido não impacta de forma negativa a experiência do utilizador.

Desta forma, decidiu-se implementar a biblioteca de CefSharp [86] e adaptá-la ao contexto da aplicação.

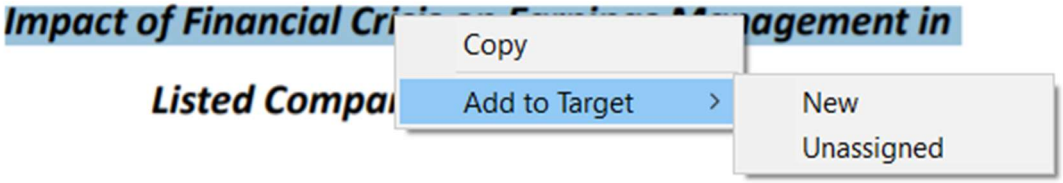


Figura 34 - Opção de inserção de dados através do navegador

Uma vantagem da utilização da seguinte biblioteca é a possibilidade de interagir com o documento web diretamente na aplicação, o que permite integrar a informação num *workspace*. O exemplo da Figura 34 demostra uma funcionalidade desenvolvida que tira o proveito desta integração. Desta forma, temos um navegador perfeitamente funcional e diretamente ligado ao *workspace* do investigador.

É de notar que a biblioteca recai sobre a licença BSD [87] que impacta o projeto de seguinte forma:

- A redistribuição do código fonte tem de manter o aviso de diretos de autor e a lista de condições que este contém.
- As redistribuições na forma binária devem reproduzir o aviso de direitos de autor e a isenção de responsabilidade na documentação e outros materiais fornecidos com a distribuição.
- Nem o Google Inc, o Chromium Embedded Framework, o nome CefSharp, assim como os nomes dos seus contribuidores podem ser utilizados para promover os produtos derivados da aplicação desenvolvida, sem permissão prévia especifica por escrito.

Ou seja, a licença BSD referida não afeta o objetivo da aplicação proposta, nem o contexto da sua utilização.

3.4.4. NodeControl

Desde o conceito e prototipagem da aplicação, a ideia de interação com os dados foi inspirada pelo Maltego. Para cumprir os requisitos de usabilidade, a informação terá de ser apresentada num grafo com as seguintes caraterísticas referidas anteriormente:

- *Drag & Drop* dos objetos de dados;
- Duplo clique no objeto para editar os dados;
- Interligar os objetos com um clique do rato;
- Seleção de múltiplos objetos de dados simultaneamente;
- *Zoom in/Zoom out*;
- Atalhos para as funcionalidades;
- Sincronizar os dados do grafo com o *workspace*;
- Diferenciar os dados utilizando cores diferentes de representação;
- Criar os dados a partir do grafo;
- Compor uma consulta utilizando os dados do grafo;
- Abrir a página web no navegador a partir do grafo;
- Manipulação do tamanho do objeto.

Muitas das caraterísticas referidas são específicas ao contexto do projeto. No entanto, as caraterísticas base de um grafo, como drag & drop, duplo clique para editar e opções de *zoom,* podem ser importados de uma fonte externa para reduzir o tempo de desenvolvimento. Para isto, o projeto inclui uma versão fortemente modificada da biblioteca Node Control [84], modificações essas que foram desenvolvidas no decorrer deste projeto.

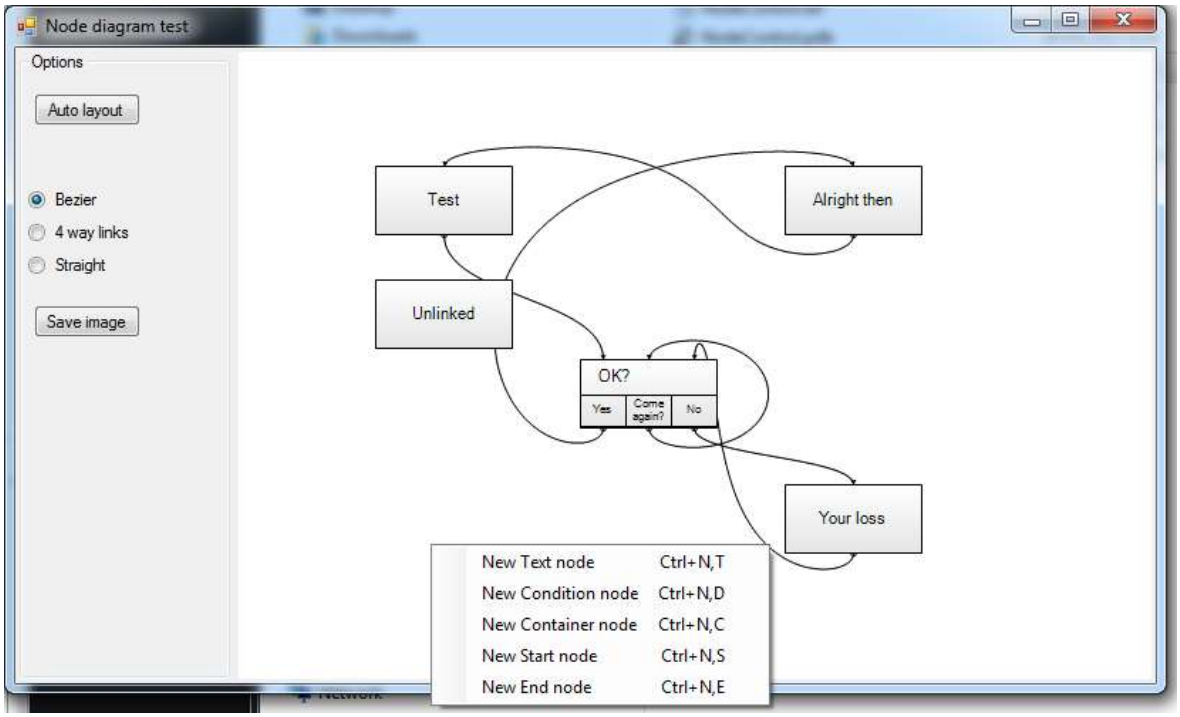


Figura 35 - Representação gráfica da versão original de Node Control [84]

Por omissão, a versão original de Node Control oferece funcionalidades básicas como drag&drop, zoom, ligação entre objetos através do rato, seleção de múltiplos objetos simultaneamente e alguns atalhos para as funcionalidades. No entanto, para as utilizar na aplicação foi necessário adaptar tudo ao contexto do *workspace*, redesenhar os objetos, criar elementos de ligação e corrigir o *layout*. Com a quantidade de modificações realizadas foi necessário criar um repositório, com o projeto modificado no GitHub [88], designado por FREE-OSINT Node Control.

Para usufruir ao máximo das funcionalidades oferecidas pelo FREE-OSINT Node Control serão aqui descritas e demonstradas as suas caraterísticas no contexto da aplicação.

- *Drag&Drop*

O mecanismo de *drag&drop* permite manipular a posição dos objetos com os dados para melhorar a construção do ambiente de investigação. Basta clicar com o rato no objeto e

arrastar para o local desejado. Esta funcionalidade foi adaptada para permitir notificar a aplicação principal das alterações de posição efetuadas e os objetos afetados, de forma a sincronizar os eventos do diagrama com o *workspace*. Um exemplo pode ser visualizado na Figura 36.

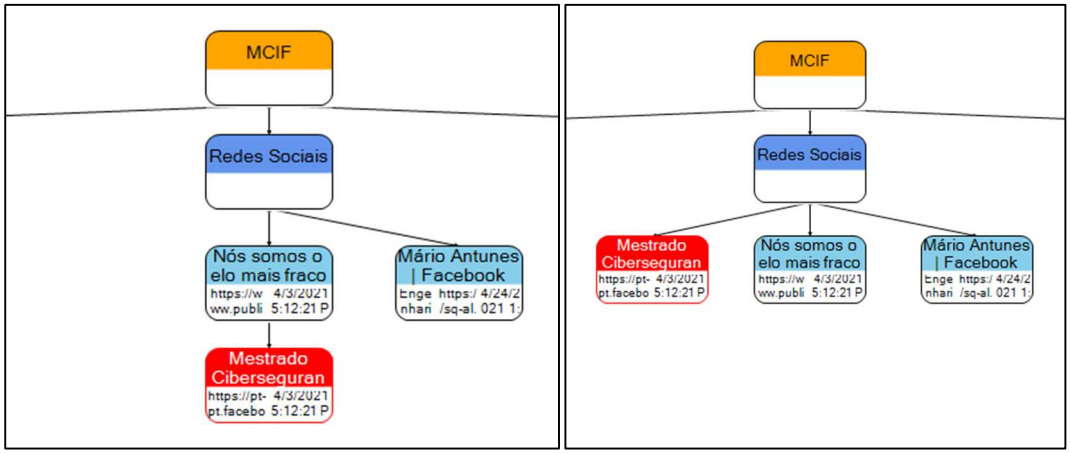


Figura 36 - *Drag & Drop* dos contentores

- Duplo clique no objeto para editar os dados

Para editar um objeto basta clicar duas vezes no objeto. A aplicação irá abrir um *pop-up* com as informações do nó escolhido. A caixa de texto superior indica o título do contentor, enquanto as caixas inferiores são elementos filho deste. Esta funcionalidade foi adaptada para notificar a aplicação principal do objeto editado e sincronizá-lo no *workspace*. Um exemplo pode ser visualizado na Figura 37.

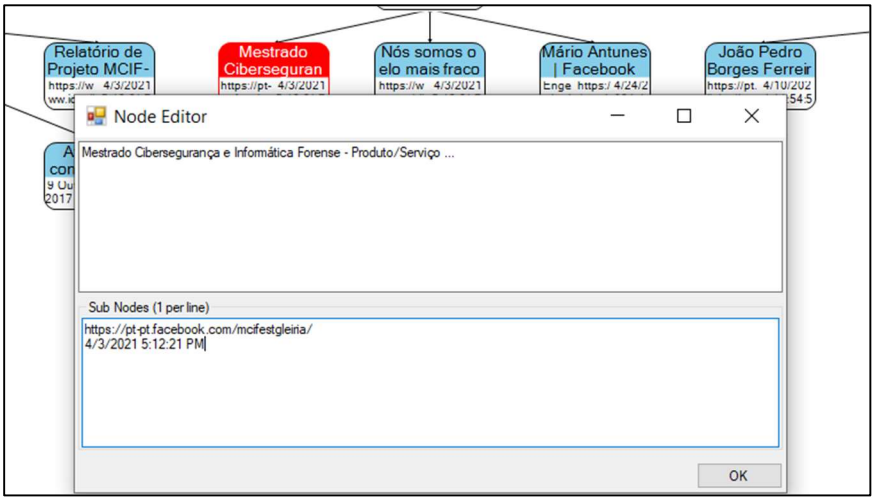


Figura 37 - Duplo clique para editar os contentores

- Interligar os objetos com um clique do rato

Para ligar os objetos, que neste caso é atribuir ao nó pai um determinado filho, basta arrastar o rato com o botão direito do objeto pai para um objeto filho pretendidos. A seguinte funcionalidade também foi adaptada com eventos de notificação para permitir sincronizar ligações entre contentores com a TreeView correspondente no workspace. Um exemplo pode ser visualizado na Figura 38.

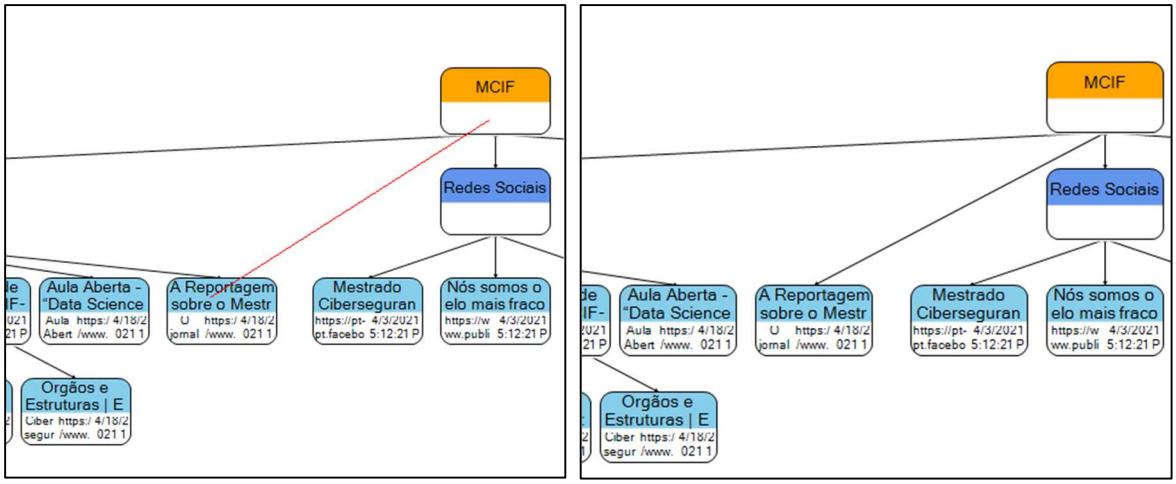


Figura 38 - Interligação dos objetos

- Seleção de múltiplos objetos de dados simultaneamente

Para selecionar muitos objetos simultaneamente basta segurar na tecla CTRL + Botão esquerdo do rato e arrastar para criar a área de objetos pretendida. Um exemplo pode ser visualizado na Figura 39.

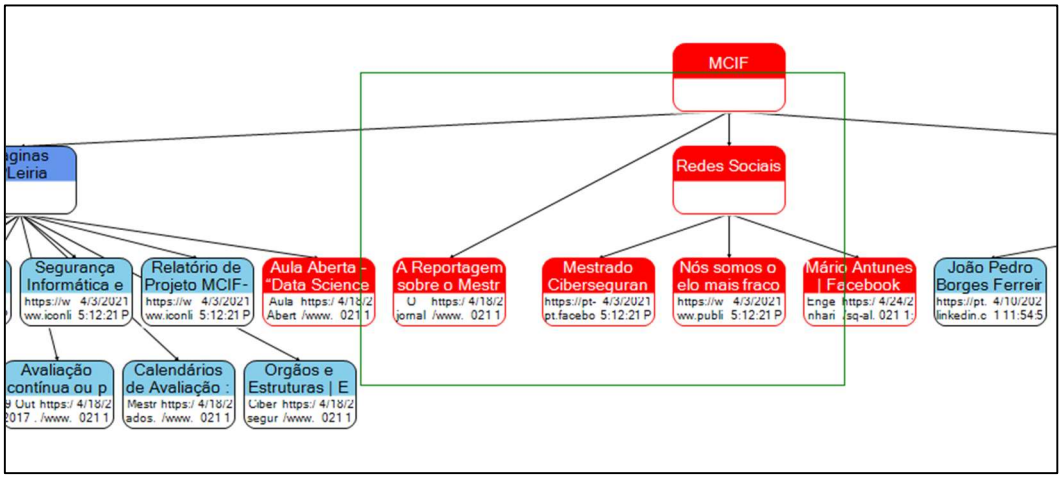


Figura 39 - Seleção de múltiplos objetos de dados simultaneamente

- *Zoom in/Zoom out*

Quando um *workspace* se torna maior, cria-se a necessidade de ampliar ou reduzir a imagem. Para este efeito é possível alterar a amplitude utilizando o atalho CTRL e a roda do rato ou fazer o gesto de ampliar/reduzir a imagem no *touchpad*. Um exemplo pode ser visualizado na Figura 40.

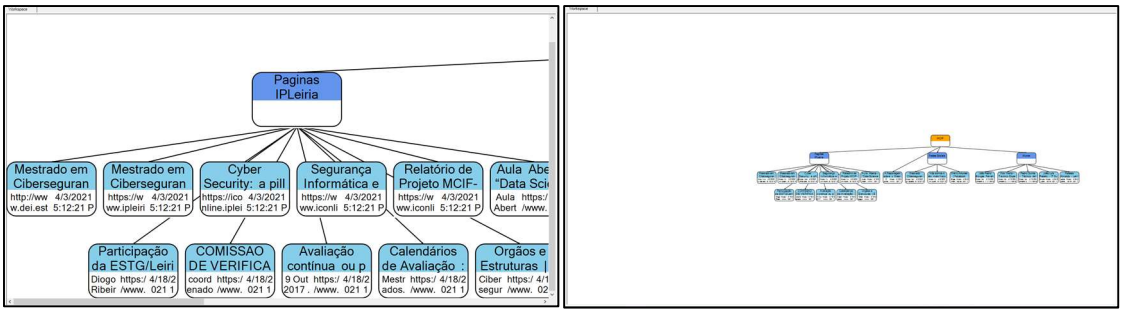


Figura 40 - *Zoom in/Zoom out*

- Atalhos para as funcionalidades

Existe um conjunto de atalhos para as funcionalidades do Node Control. Todos estes são apresentados em menus de contexto. Foram adaptados e adicionados novos atalhos que permitem interagir ao nível do *workspace*. É permitido através dos atalhos: Criar objetos Target e Intel, modificar a cor do objeto, compor uma nova consulta, abrir o objeto num navegador e, no caso de necessidade, também o remover. Um exemplo pode ser visualizado na Figura 41.

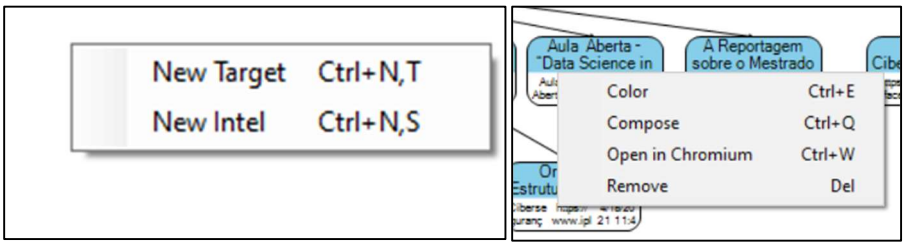


Figura 41 - Atalhos para as funcionalidades

- Sincronizar os dados do grafo com o workspace

Uma vantagem da utilização do Node Control é o facto deste estender a classe "Component" na *framework* de .NET, o que permite implementá-lo numa Windows Forms diretamente. Com um conjunto de EventHandlers, todas as alterações feitas no diagrama notificam a aplicação principal, atualizando assim o *workspace*, os itens associados e vice versa.

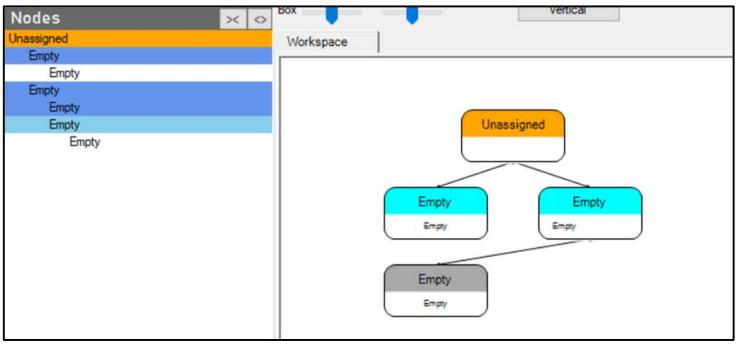


Figura 42 - Sincronização dos dados do grafo com workspace

Todas as alterações realizadas que necessitem de notificar a aplicação principal, utilizam a classe DiagramEventArgs desenvolvida como ponte entre o diagrama e a janela principal, onde indicam numa lista, os objetos selecionados e numa propriedade do tipo Operation_Type, a operação realizada. Uma representação desta sincronização pode ser visualizada na Figura 42. No momento são implementadas 7 operações de ligação: *add*, *edit*, *remove*, *link*, *drag*, *compose*, *color* e *open URL*.

- Diferenciar os dados utilizando cores diferentes de representação

Para permitir agrupamento visual dos dados é permitido alterar a cor do contentor de dados. Basta selecionar o contentor com o botão direito do rato e escolher a opção "Color", ou utilizar o atalho CTRL + E. Um exemplo pode ser visualizado na Figura 43.

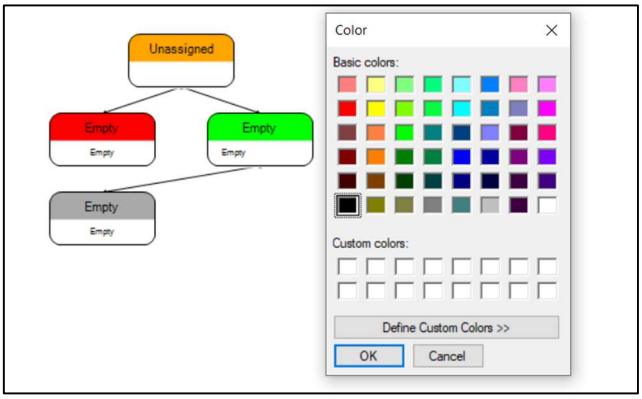


Figura 43 - Diferenciação dos dados através de cores

- Criar os dados a partir do grafo

Para o caso de investigador ter obtido informação a partir de fontes externas e as querer incluir no *workspace*, foi adaptada a funcionalidade de criar contentores do tipo Target e Intel, através do clique no espaço em branco com o botão direito do rato ou utilizando os

atalhos especificados (CTRL+N+T ou CTRL+N+S). Um exemplo pode ser visualizado na Figura 41.

- Compor uma consulta utilizando os dados do grafo

Para continuar a pesquisa através da informação existente, foi desenvolvida a funcionalidade de compor uma consulta e é acessível através do atalho CTRL+Q ou botão direito do rato e seleção da opção "Compose Query". Isto irá invocar o processo de criação da consulta composta, que consiste em juntar o texto do contentor alvo com o selecionado e, em caso de necessidade, adicionando contentores intermédios. Caso seja necessário juntar informação de contentores vizinhos, basta carregar em "Yes" na questão "Query generated, do you wish to edit?", que irá invocar a janela de construção da consulta. Um exemplo pode ser visualizado na Figura 44.

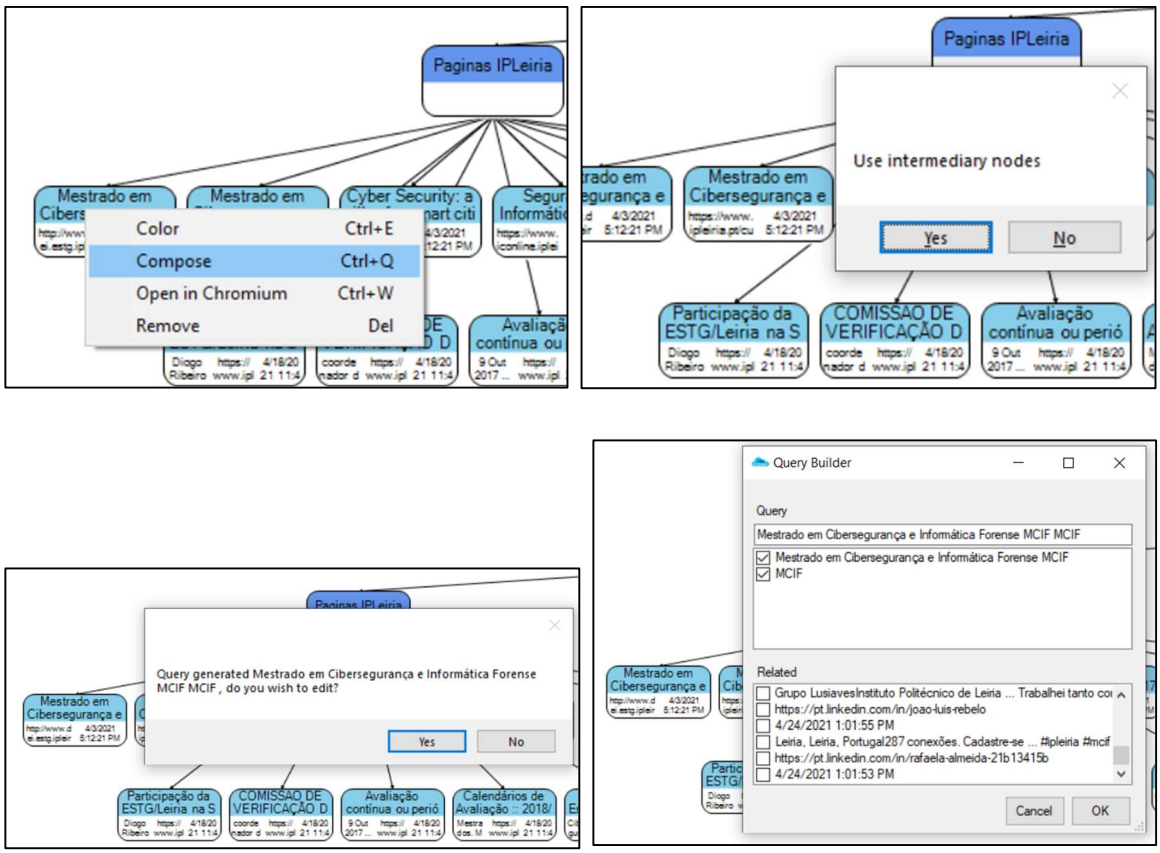


Figura 44 - Composição do termo de consulta utilizando os dados do grafo

- Abrir a página web no navegador a partir do grafo

Como referido anteriormente, a classe que interliga o diagrama com a aplicação principal permite passar qualquer tipo de informação entre eles, o que possibilita, no caso de um contentor com uma hiperligação, abri-la no navegador embutido da aplicação principal. Para

usufruir desta funcionalidade basta carregar com o botão direito do rato no contentor com o endereço e escolher “Open in Chromium” para abrir um separador, ou utilizando o atalho CTRL+W. Um exemplo pode ser visualizado na Figura 45.

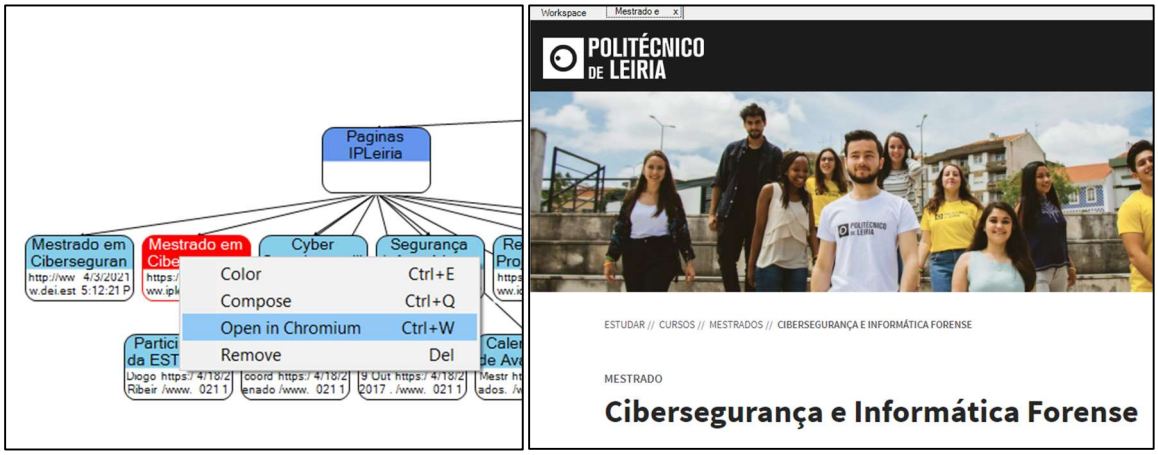


Figura 45 - Abrir a página web no navegador a partir do grafo

- Manipulação do tamanho de objetos

Para dispositivos com densidade de pixéis do monitor reduzida pode surgir a necessidade de modificar o tamanho dos contentores. De modo a permitir ajustar o tamanho dos contentores foi criado o mecanismo que liga o comprimento e a largura destes a valores em *sliders*. Para usufruir desta funcionalidade são utilizados os *sliders* no canto superior direito. Um exemplo pode ser visualizado na Figura 46.

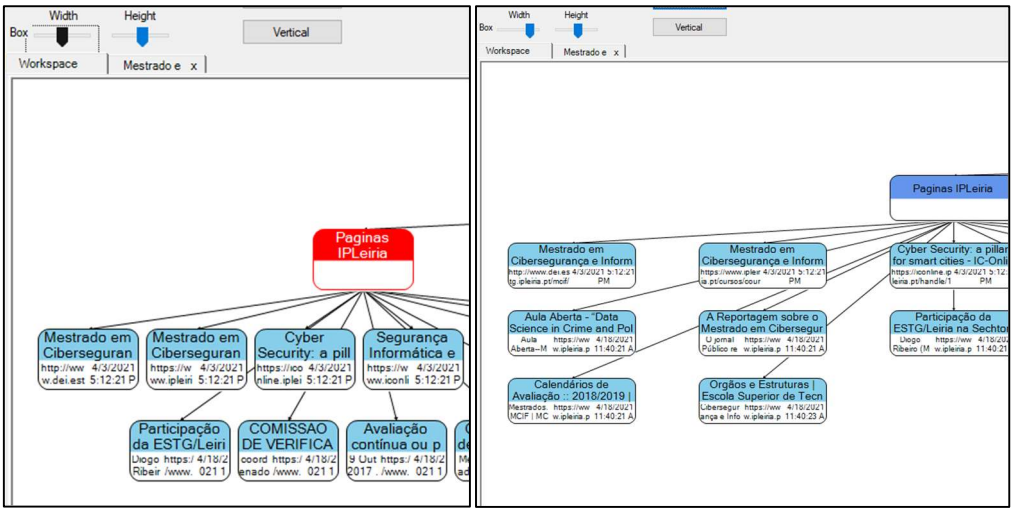


Figura 46 - Manipulação do tamanho dos contentores

3.4.5. FREE-OSINT Google Custom Search

O seguinte módulo foi desenvolvido para mostrar as capacidades de um módulo de pesquisa e usufrui de funcionalidades oferecidas pela Google Custom Search API [39]. Como foi referido anteriormente, o serviço REST necessita de três parâmetros nos seus pedidos que são os seguintes:

- Chave API – identifica a aplicação, necessita de uma conta google existente.
- Programmable Search Engine ID – identifica o motor de pesquisa a utilizar, associado a chave API.
- Query de pesquisa – expressão utilizada para a pesquisa.

Na Figura 47 encontra-se um exemplo de pedido efetuado para a Custom Search API, com os três parâmetros, em que, a *key* representa a chave API, *cx* representa a Programmable Search Engine ID e *q* representa a query.

```
GET https://www.googleapis.com/customsearch/v1?key=INSERT_YOUR_API_KEY
&cx=017576662512468239146:omuauf_lfve
&q=lectures
```

Figura 47 - Exemplo de pedido realizado para Custom Search API

A resposta devolvida depende do pedido, podendo ou não conter resultados num vetor de itens em formato JSON.

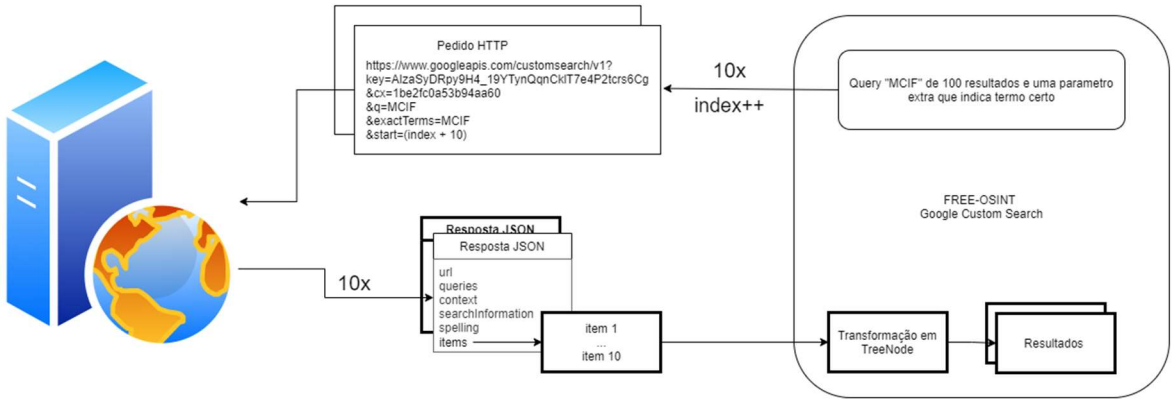


Figura 48 - Processo de realização da consulta

No nosso caso, é devolvida a resposta em formato JSON. Todas as queries realizadas são convertidas num pedido HTTP, indicando, ou não, parâmetros extra separados por '&'. A resposta do servidor contém um conjunto de informações que podem ser consideradas úteis,

como o número total de resultados possíveis, o tempo despendido para a realização da pesquisa, sugestões de correção de escrita e os parâmetros de acesso à próxima página. A importância do último ponto vem da necessidade de realizar múltiplas consultas para obter mais de 10 resultados, devido a configuração deste. Cada consulta realizada permite obter apenas 10 resultados de cada vez, necessitando a realização da mesma consulta novamente, com um parâmetro extra indicando o índice de resultados. Por exemplo, sendo o índice inicial de 1 mais 10 dos resultados obtidos, ou seja, a próxima consulta terá de indicar o índice de 11 para a obtenção dos resultados. Os resultados da pesquisa são localizados no vetor de itens da resposta. Cada resultado do vetor de itens contém um conjunto de informações, dos quais os mais relevantes são:

- Title – título da página web.
- Link – endereço URL da página.
- Snippet – um fragmento da informação da página.
- Pagemap->metatags – meta dados associados ao resultado, que variam entre resultados.

Ao fim de receber a resposta do servidor, o JSON é convertido em classes designados por Result. Estes são utilizados para a representação visual dos resultados, para que este módulo seja interativo. Os resultados então são armazenados numa lista e apresentados ao utilizador. Todo este processo é ilustrado pela Figura 48. Caso a pesquisa tenha sido efetuada através do método Search, que necessita dos resultados no formato universal para os módulos, os resultados são transformados num TreeNode com o nome do módulo na raiz.

Para além de WinForms dedicados a interação com o módulo existem classes que contribuem para a lógica do funcionamento deste. O diagrama de classes da Figura 49 demonstra a relação entre estes.

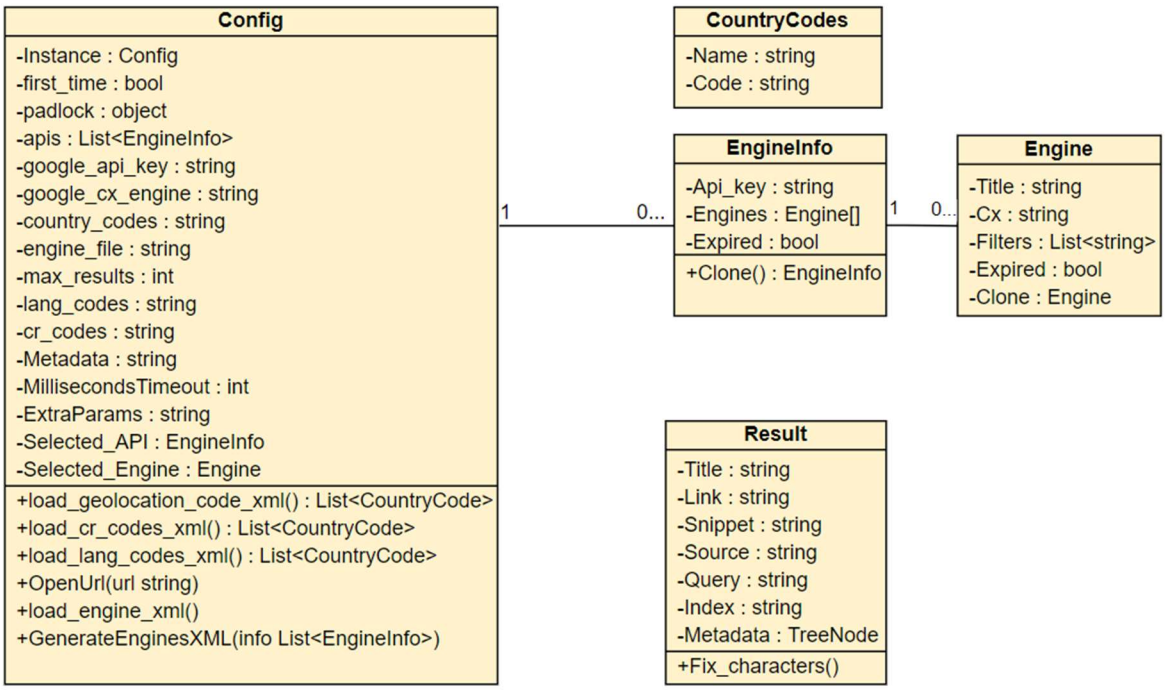


Figura 49 - Diagrama de classes do módulo FREE-OSINT Google Custom Search.

Para personalizar a pesquisa existem dois mecanismos a ter em conta. O primeiro é um sistema de parâmetros, que na documentação da API são divididos em dois grupos: parâmetros de API, que definem as propriedades da pesquisa, como expressões, número de resultados, linguagem e etc; e parâmetros standard, que definem aspetos técnicos dos pedidos, como chave API ou do *engine*. Todos os parâmetros terão de ser codificados para o formato URL. No momento existem 31 parâmetros dedicados a API e 9 parâmetros standard [89]. O segundo mecanismo de personalização da pesquisa são as search engines dedicadas. Após obtenção da chave API é necessário criar um *custom search engine*, para que o seu ID seja utilizado nos pedidos.

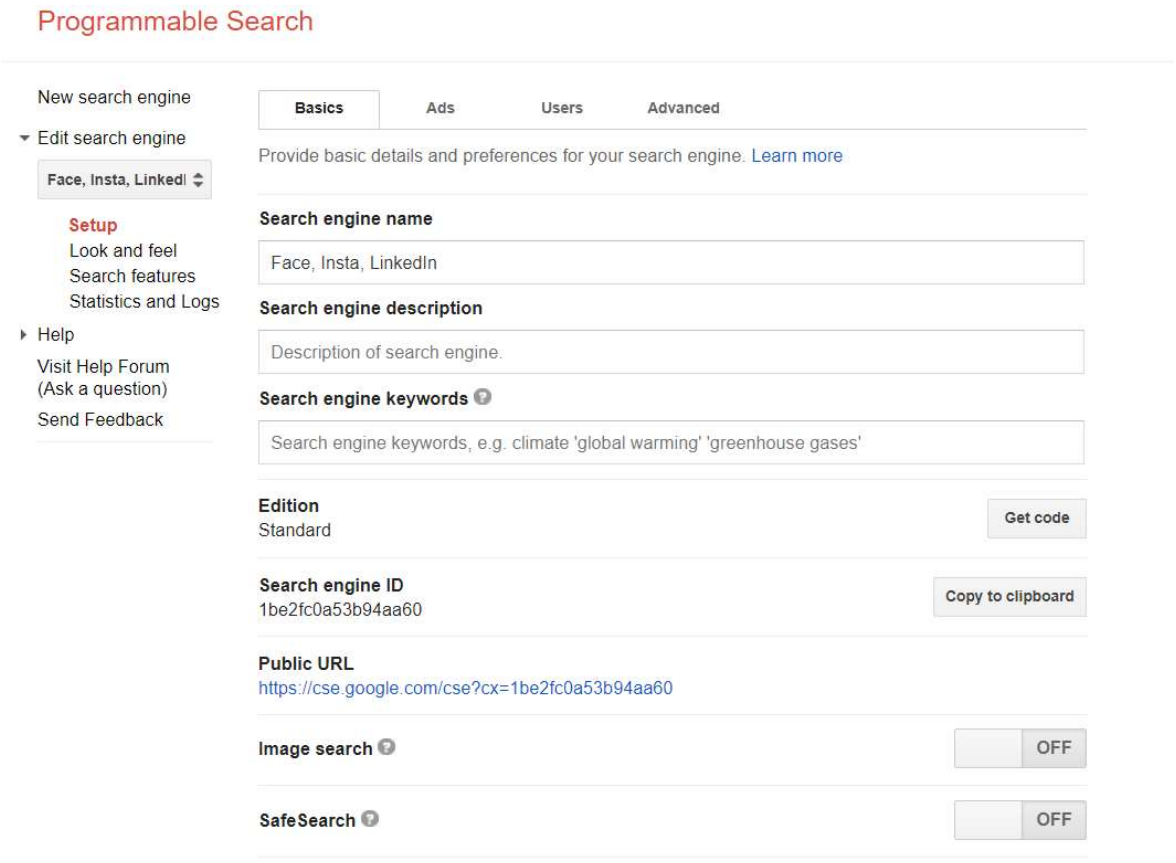


Figura 50 - Definições de *engine*

A gestão dos *custom search engines* é feita no painel de controlo do utilizador [90]. O aspeto de personalização no que se refere aos *engines*, vem de configurações feitas a estas, como: restrições a pesquisas numa região específica; filtragem de páginas web; exclusão de resultados de imagens; e etc. Um exemplo do painel de controlo é ilustrado na Figura 50.

A gestão das chaves API e os seus *engines* é feita através de um ficheiro XML, designado por "engines.xml", exemplificado na Figura 51. Cada chave API é indicada pelo atributo API do elemento google_api. Neste também é possível encontrar elementos com o nome *engine* que, para além do título e o código cx, também possuem a lista de filtros, que serão apresentadas nas próximas páginas.

```
<document>
  <google_api api="AIzaSyDRpy9H4_19YTynQqnCklT7e4P2tcrs6Cg">
    ...
  </google_api>
  <google_api api="AIzaSyCSCObMTpG_2VzZ5jbwlAdVrQFRXU3-vLI">
    <engine>
      <Title>Reddit and Anything</Title>
      <cx>bcd655709061cc99e</cx>
    </engine>
    <engine>
      <Title>Ipleiria and Anything</Title>
      <cx>ef20f4b123d7fa0e5</cx>
    </engine>
  </google_api>
</document>
```

Figura 51 - Ficheiro engines.xml

Caso o ficheiro não se encontre na raiz, é assumido que é a primeira interação do utilizador com o módulo, executando em primeiro lugar a janela dos primeiros passos, ilustrada pela Figura 52.

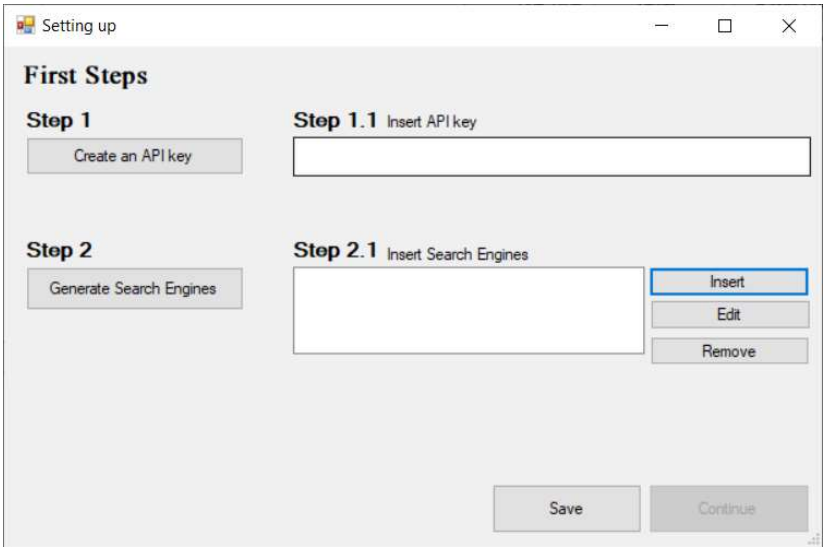


Figura 52 - Janela dos primeiros passos

O objetivo é facilitar ao utilizador a configuração inicial das chaves através da seguinte lógica. O primeiro passo é a opção “Create an API key”, que leva para a página de *overview* da API, Figura 53, onde se encontra a opção “Get a Key”. Após obtenção da chave basta indicá-la no campo “Insert API key”.

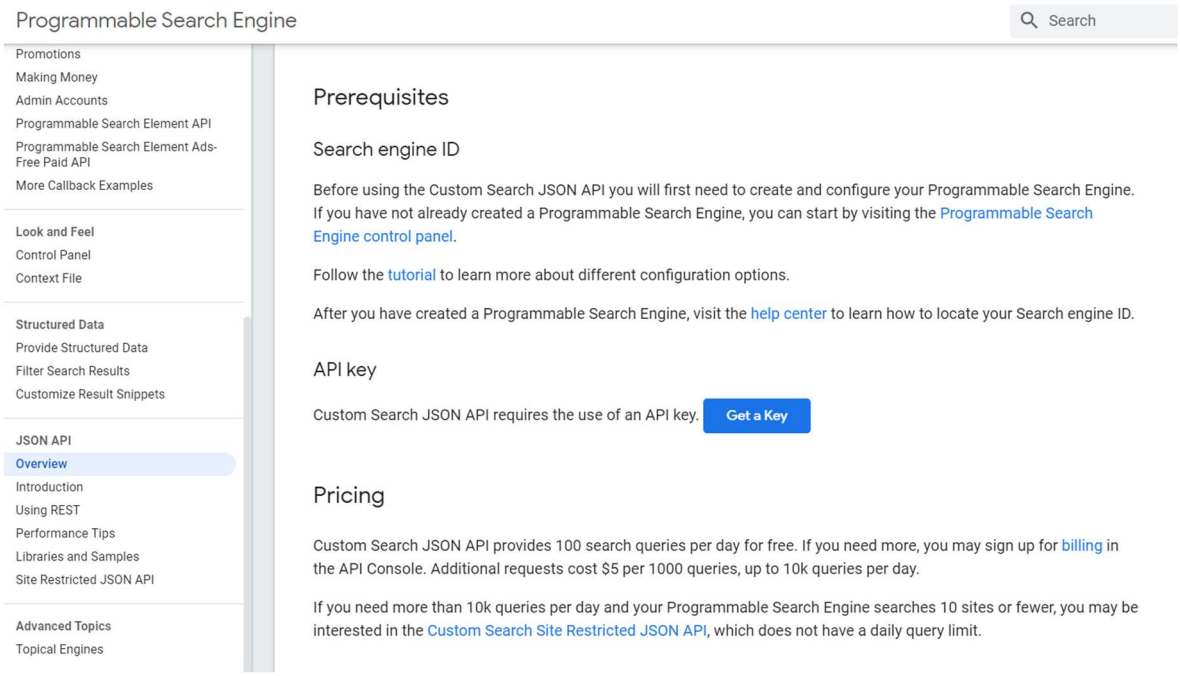


Figura 53 - Página de overview da JSON API

O próximo passo será a criação do *engine* e obter a sua chave através da opção "Generate Search Engines", que leva o utilizador para o painel de Search Engines, exemplificado na Figura 54.

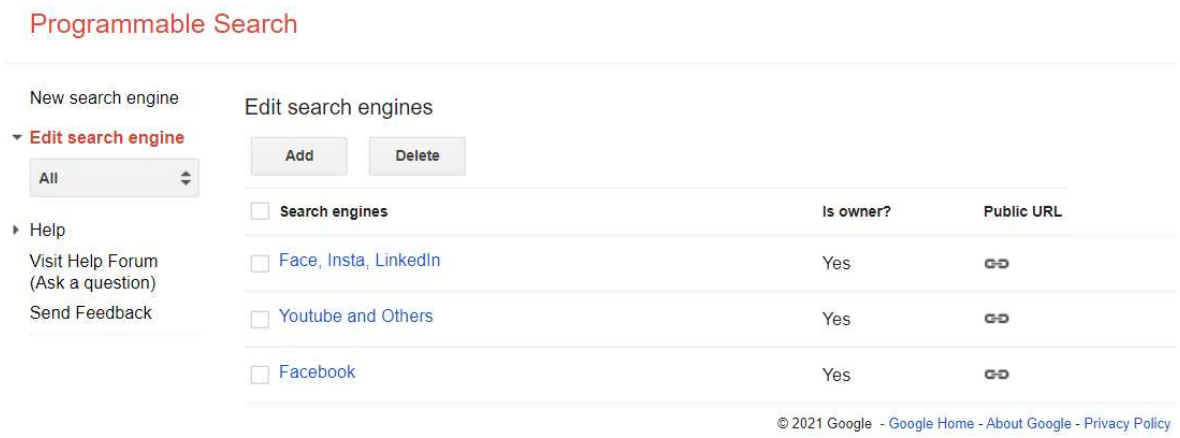


Figura 54 - Painel de Search Engines

Os *engines* criados deverão de ser transportadas para a janela dos primeiros passos através da opção "Insert" do passo 2.1. Isto irá invocar a janela de inserção de um *engine*, ilustrada pela Figura 55. Aqui, o utilizador decide o nome que deseja atribuir o *engine* e, caso tenha filtros de páginas web, inseri-los. Os filtros são um elemento opcional e apenas têm utilidade em *engines* que incluem vários domínios. Cada filtro inserido na lista de filtros irá criar um

subconjunto de resultados com o domínio indicado e apresenta uma ação extra para acesso rápido para cada na secção “Engine filters” do lado esquerdo da janela principal, ilustrado pela Figura 56.

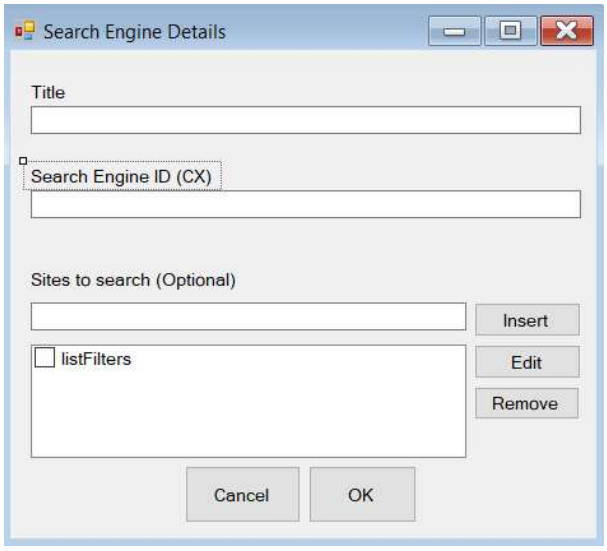


Figura 55 - Janela de inserção de um *engine*

Ao concluir a inserção dos *engines* concluem-se os primeiros passos e guardam-se as configurações através da opção “Save”. Caso os dados tenham sido preenchidos corretamente será desbloqueada a opção “Continue”, que leva para a janela principal do módulo, demonstrada pela Figura 56.

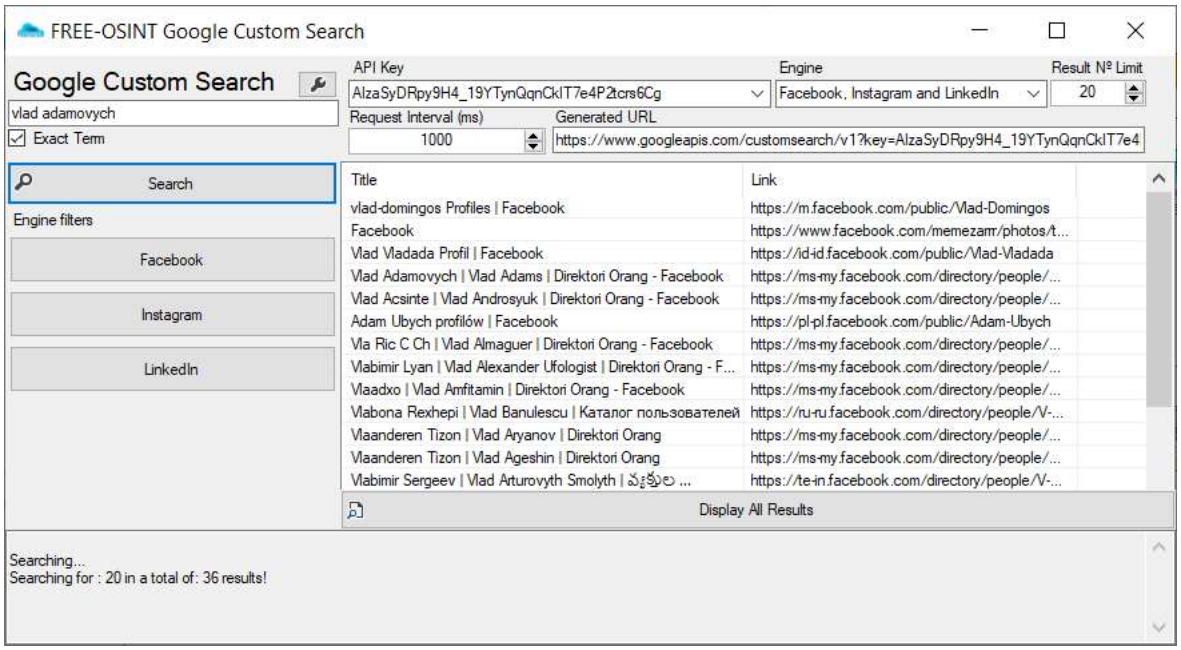


Figura 56 - Janela principal do módulo FREE-OSINT Google Custom Search.

A interface da janela principal tenta manter a estrutura simples e intuitiva. O termo da consulta é inserido e executado no canto superior esquerdo e os resultados aparecem no centro em forma de lista. Os restantes elementos são opcionais. As opções de chaves API e Engine são automaticamente preenchidas a partir do ficheiro, a primeira sendo selecionada para utilização.

A opção "Result Nº limit" indica a quantidade de resultados máximo pretendido. O valor de "Request Interval (ms)" indica o tempo de espera entre cada pedido efetuado a API. Como foi referido anteriormente, cada resposta da Custom Search API devolve apenas 10 resultados, como ilustrado pela Figura 57, sendo necessário efetuar de novo um pedido, indicando um novo índice sendo este, o valor antigo de índice com um acréscimo de 10, caso se pretende obter mais de 10 resultados. Este valor de intervalo é necessário para situações em que a API é sobrecarregada e responde com o código "(429) Too Many Requests.".

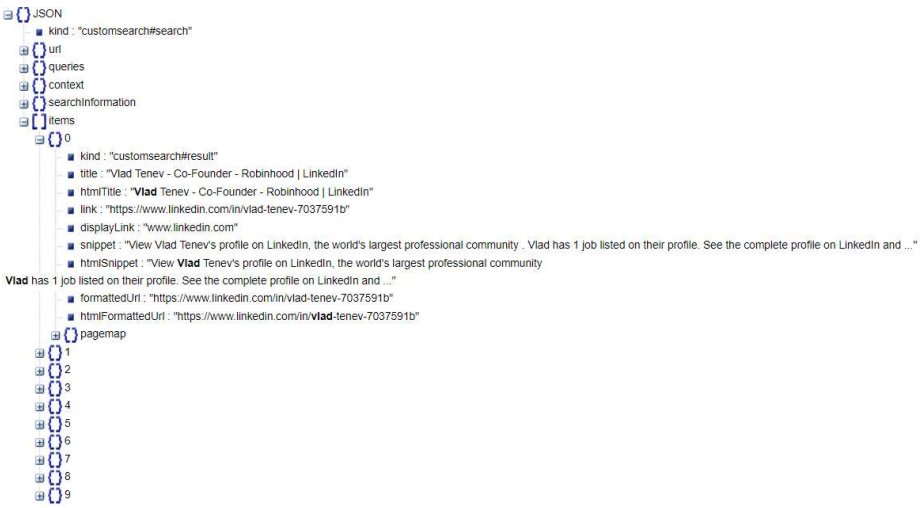


Figura 57 - Resposta devolvida pelo Custom Search API

O campo "Generated URL" mostra o URL gerado pela consulta para que seja possível reutilizá-lo num browser, caso o utilizador pretenda visualizar a resposta numa ferramenta externa.

A CheckBox com o nome "Exact Term" indica a necessidade do *engine* pesquisar por ocorrência do termo de consulta, exatamente como esta se encontra no formulário e não como um conjunto de palavras independentes.

As opções do campo "Engine Filters" são criados a partir dos filtros indicados na Figura 55. Utilizando a Figura 56 como exemplo, é utilizada uma *engine* que pesquisa apenas em páginas de Facebook, LinkedIn e Instagram, estes sendo indicados na opção de filtros do

engine. Isto permite filtrar a informação automaticamente, em que cada opção abre a janela de resultados, ilustrada pela Figura 58, com resultados do serviço pretendido.

Na parte inferior da janela principal do módulo encontra-se o painel de eventos. Neste painel é exibida a informação relacionada com os eventos ocorridos no módulo, a quantidade de resultados recolhidos num total possível ou as mensagens de erro devolvidas pela API.

A opção "Display All Results", como o nome indica, apresenta todos os resultados numa janela de resultados, ilustrada pela Figura 58, para possível visualização ou inserção no *workspace* da aplicação.

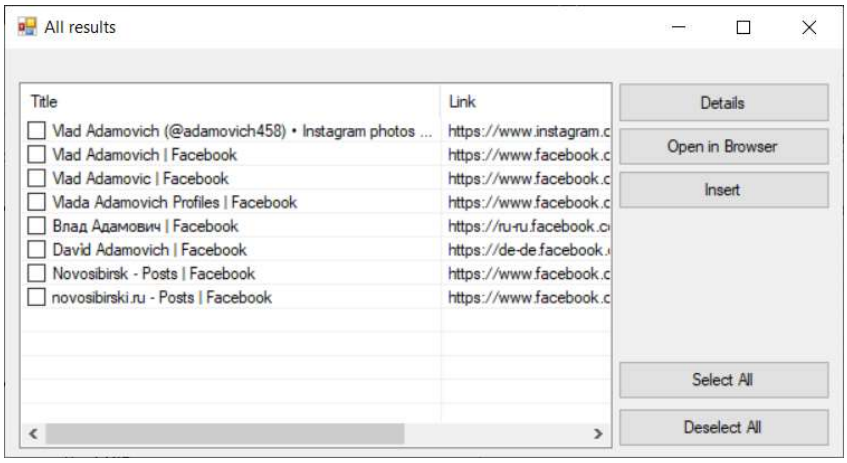


Figura 58 - Janela de resultados

Na janela de resultados é possível interagir com a informação de diversas formas. Caso o utilizador pretenda visualizar a informação a partir da aplicação utiliza-se a opção "Details", que invoca o evento da biblioteca partilhada OPEN_URL, abrindo assim a hiperligação no navegador da aplicação principal. Também é possível visualizar a informação num navegador do sistema operativo através da opção "Open in Browser". A opção "Insert" invoca o evento INSERT da biblioteca partilhada para que seja possível inserir a informação no *workspace*, através da janela de seleção do alvo da aplicação principal, ilustrado pela Figura 59.

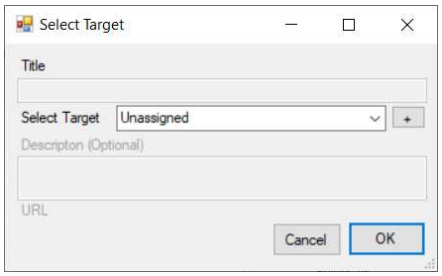


Figura 59 - Janela de seleção do alvo

A opção de parametrização da pesquisa, ilustrada na Figura 60, que se encontra no canto superior esquerdo da janela principal, abre a janela de parametrização da pesquisa. Nesta janela encontram-se os principais parâmetros utilizáveis indicados na documentação [91].

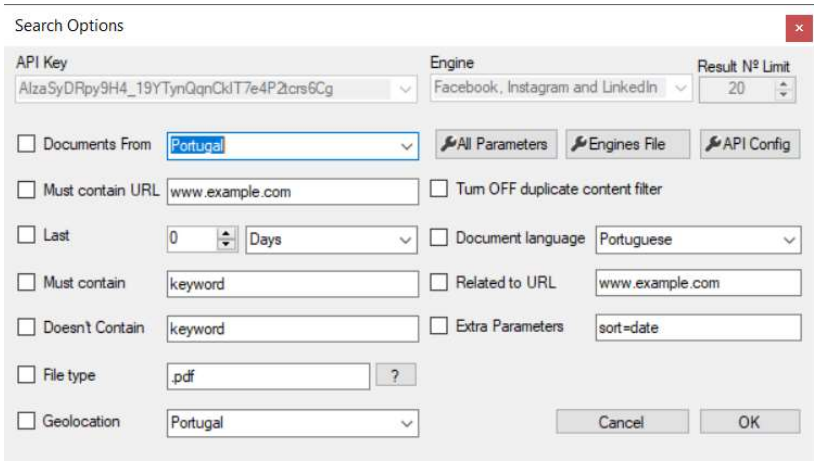


Figura 60 - Janela de parametrização da pesquisa

A janela de parametrização da pesquisa também é invocada pelo método "Configure", na implementação de interface IConfigurable_module. Os parâmetros existentes de momento são seguintes:

- "Documents From" – indica o país onde se encontram os resultados.
- "Must contain URL" – indica o URL que os resultados devem ter.
- "Last" – indica o quão recente é a informação recolhida.
- "Must contain" – indica a necessidade de resultados conterem a palavra inserida.
- "Doesn't contain" – indica a necessidade de resultados não conterem a palavra inserida.
- "File type" – indica o tipo de resultado pretendido, dentro da lista de possíveis tipos indicados na opção com "?" em anexo.
- "Geolocation" – indica a localização do dispositivo que realiza a pesquisa.
- Turn OFF duplicate contente filter – desativa o algoritmo de filtragem de resultados com conteúdo duplicado.
- "Document language" – indica a linguagem das páginas de resultados.
- "Related to URL" – indica o endereço com que os resultados deverão estar associados.
- "Extra Parameters" – indica parâmetros extra que não se encontram na janela de parametrização de pesquisa.

Existem ainda três opções no canto superior direito para configurações extra. A opção "All Parameters" leva o utilizador para a página dos parâmetros, para que este possa visualizar todos os parâmetros aceites pela API. A opção "API Config" leva o utilizador para a página de criação/edição dos *engines*. A opção "Engines File" irá invocar a janela de configuração das chaves, ilustrado na Figura 61.

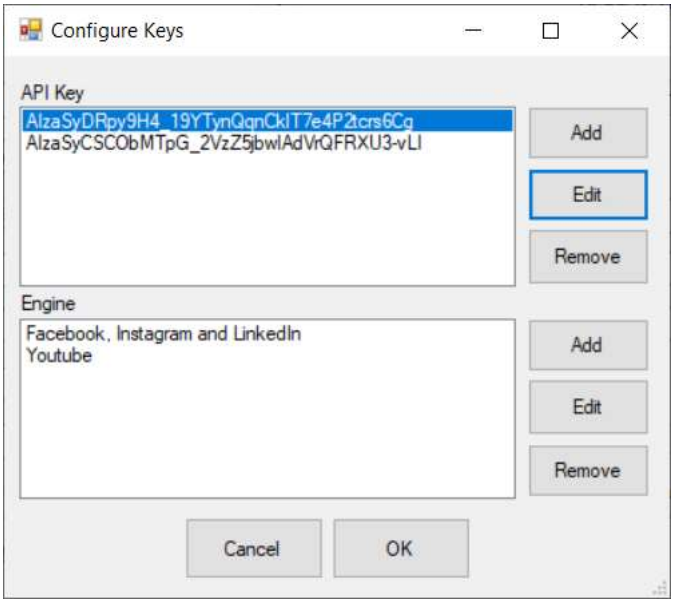


Figura 61 - Janela de configuração das chaves

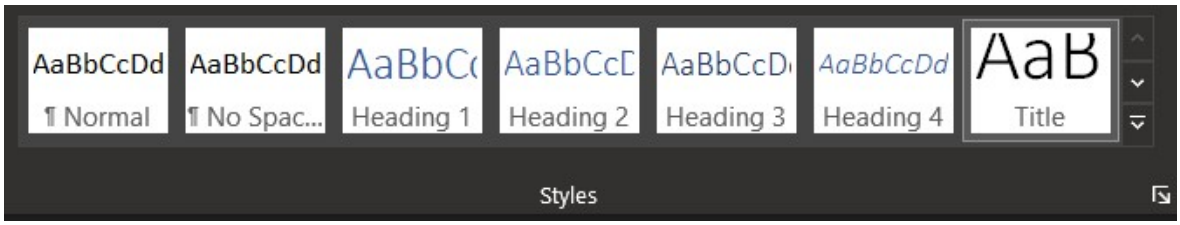
Na janela de configuração das chaves é possível editar o ficheiro "engines.xml" a partir da aplicação, por forma a permitir situações de atualização ou criação de novos *engines* ou mudanças/criações de chaves API.

Atualmente, o módulo faz leitura de quatro ficheiros de XML diferentes. Estes são:

- engines.xml – contém chaves da API e os seus respetivos *engines* e filtros;
- gl-codes.xml – contém códigos utilizados pelo campo Geolocation. Exemplo: gl=pt;
- lang_codes.xml – contém códigos utilizados pelo campo Document language. Exemplo: lr=lang_pt;
- cr-codes.xml – contém códigos utilizados pelo campo "Documents from". Exemplo: cr=countryPT.

3.4.6. FREE-OSINT Report Builder

O módulo de construção de relatórios é utilizado pela aplicação para construir relatórios, implementando desta forma a interface IReport_module. O método "GenerateDocument" da interface recolhe a lista de nós selecionados e percorre de forma recursiva toda a informação, constrói um documento DOC ou DOCX, e atribui para cada nível um estilo, para que seja possível distinguir cada um deles. Neste caso os níveis definidos são, como a Figura 62 indica, os primeiros cinco estilos "Heading" desenvolvidos pelo Microsoft Word. Os nós que não têm filhos são construídos usando o estilo de texto normal. Um exemplo do documento final construído pode ser observado na Figura 63.




```
preset_styles = new WdBuiltinStyle[5];
            preset_styles[0] = WdBuiltinStyle.wdStyleHeading1;
            preset_styles[1] = WdBuiltinStyle.wdStyleHeading2;
            preset_styles[2] = WdBuiltinStyle.wdStyleHeading3;
            preset_styles[3] = WdBuiltinStyle.wdStyleHeading4;
            preset_styles[4] = WdBuiltinStyle.wdStyleHeading5;
```

Figura 62 - Estilos definidos para os níveis de nós

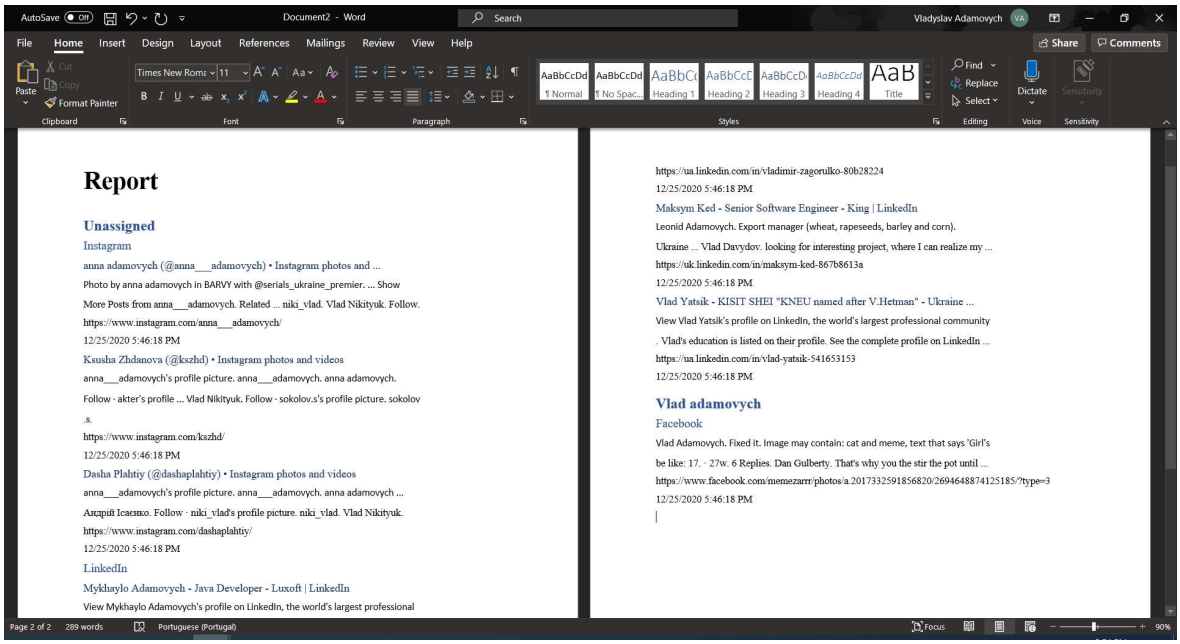


Figura 63 - Documento construído

3.4.7. Adicionar um novo módulo

Devido a modularidade da aplicação, será apresentado, em formato de tutorial, os passos para o desenvolvimento e edição de um novo módulo à aplicação FREE-OSINT. Será desenvolvido um módulo simples de processamento que terá o seguinte objetivo: extrair o primeiro e o último nome do individuo e caso seja possível, o endereço para a foto a partir dos metadados de perfis de LinkedIn obtidos a partir do modulo FREE-OSINT Google Custom Search API.

3.4.7.1. Preparar o ambiente de trabalho

O primeiro passo será escolher o template que se mais adequa ao módulo a desenvolver. Neste caso, não existe a necessidade de interagir com o módulo e uma Console App será suficiente, ilustrada na Figura 64.

O nome do seguinte modulo será "FREE_OSINT_LinkedIn_Metadata_Extractor" e a *framework* a utilizar será .NET Framework 4.7.2.

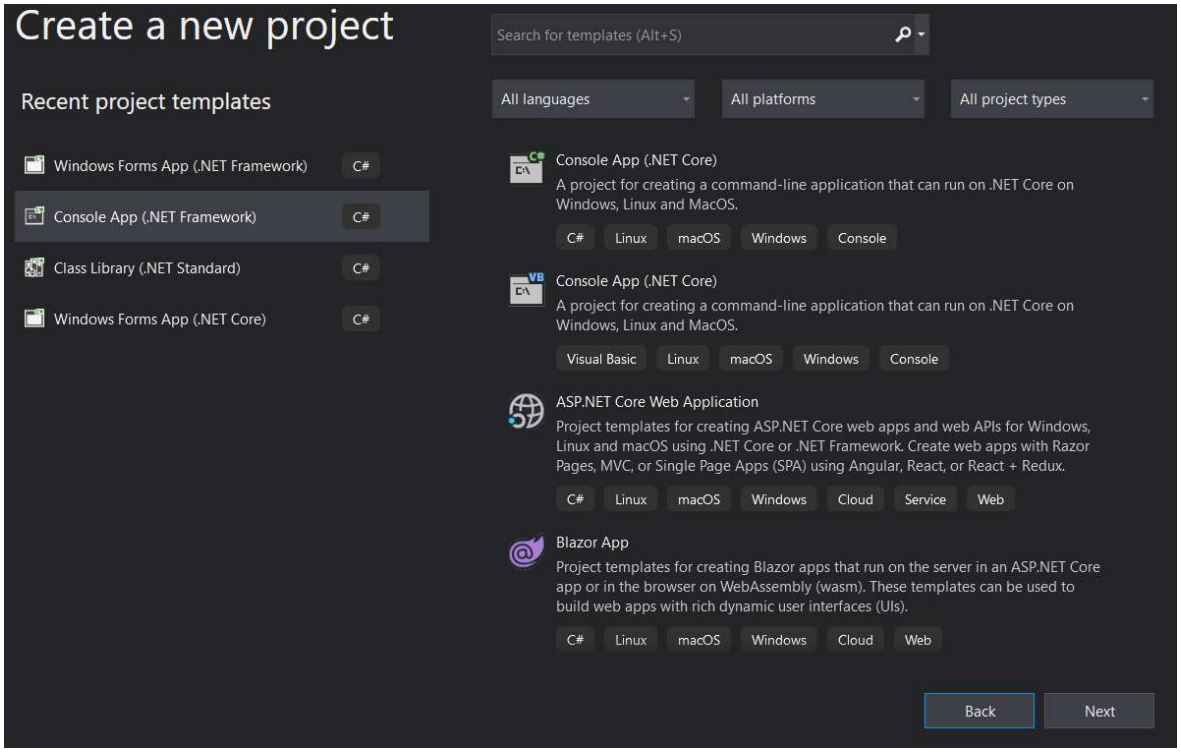


Figura 64 - Janela de criação de projeto

De seguida, será necessário obter a última versão da biblioteca partilhada que pode ser obtida dentro da diretoria "Releases" no GitHub. A versão que vamos utilizar neste exemplo é a única disponível no momento: github.com/Nekkilodeon/FREE-OSINT/releases/tag/1.0.0

O ficheiro DLL obtido pode ser colocado na diretoria da solução criada para um acesso facilitado.

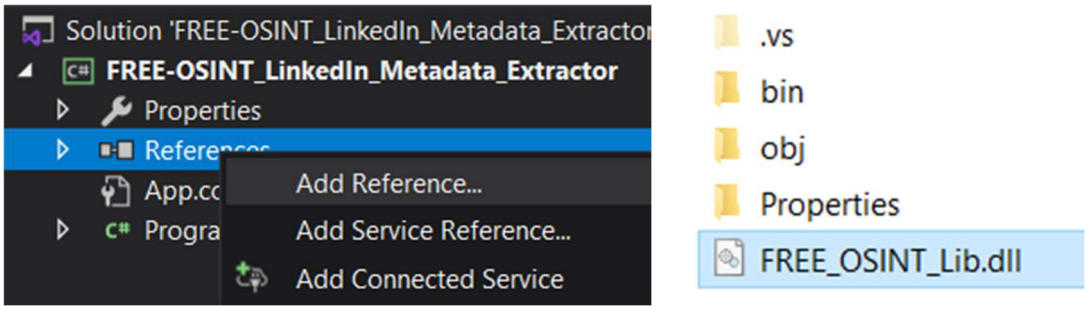


Figura 65 - Adicionamento da referência

Nas referências do programa adicionar o ficheiro, ilustrado na Figura 65, através de:

- References->Add Reference->Browse->FREE_OSINT_Lib.dll
- No código adicionar “using FREE_OSINT_Lib”;

3.4.7.2. Implementar as Interfaces

Com a biblioteca referenciada, o próximo passo passa por implementar as interfaces relevantes. Neste caso são as interfaces IGeneral_module, porque todos os módulos devem implementar essa seguinte interface, e a IProcessing_module, porque o módulo a desenvolver é do tipo processamento. O passo é ilustrado na Figura 66.

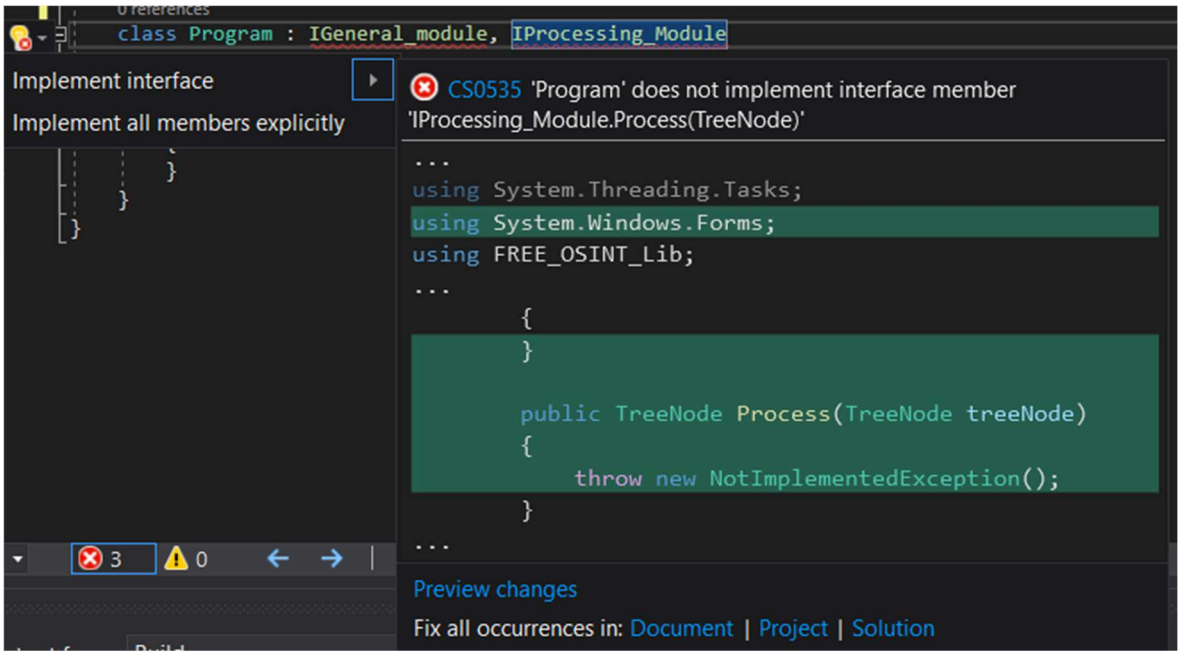


Figura 66 - Implementação das interfaces

3.4.7.3. Facilitar o processo de *debugging*

Para que os testes e *debugging* sejam mais fáceis de realizar é recomendada mudança da diretoria de Build e da aplicação de Debug, ilustrados pelas Figura 67 e Figura 68. Para o Build basta indicar a diretoria dos módulos da aplicação FREE-OSINT, juntando o nome do módulo, através do seguinte passo:

- Properties->Build->Output path
    - FREE-OSINT\modules\"Nome do modulo"

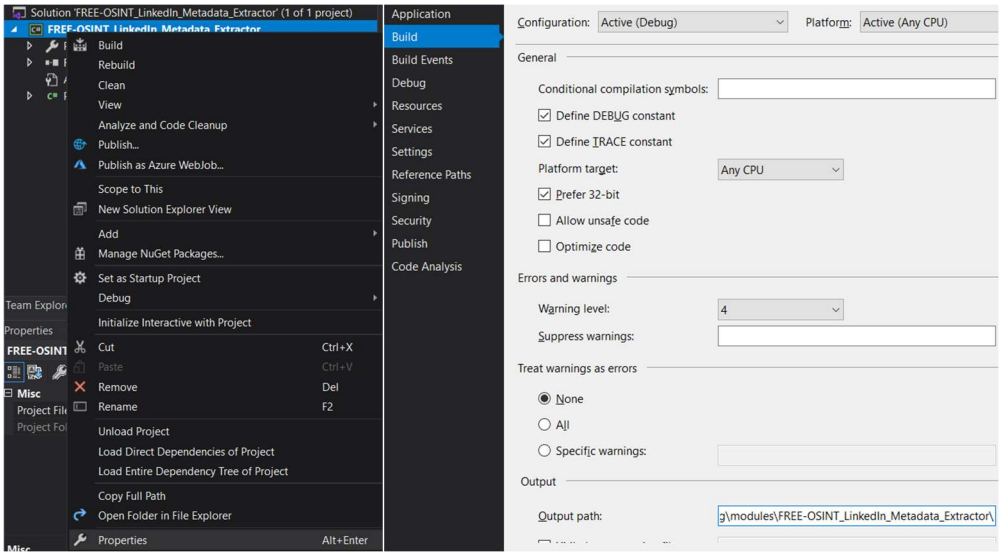


Figura 67 - Mudança da diretoria de Build

Para a aplicação de Debug fazer os seguintes:

- Properties->Debug-> Start external program
    - "Diretoria da aplicação principal\FREE-OSINT.exe
- Properties->Debug->Working directory
    - Diretoria da aplicação principal

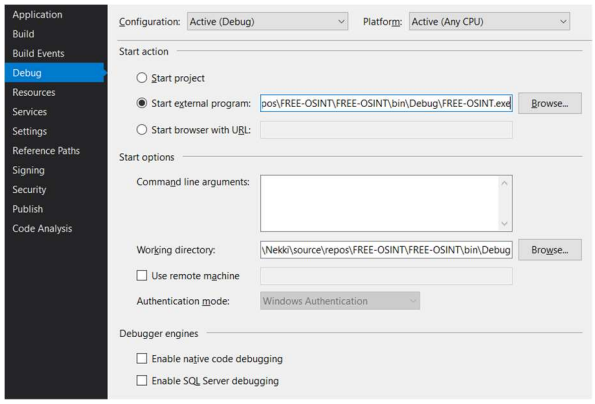


Figura 68 - Mudança da aplicação para Debug

A partir destes passos, sempre que é realizado Rebuild do módulo, este é automaticamente atualizado na diretoria FREE-OSINT\modules. Ao iniciar o processo de Debugging, executará a aplicação principal, permitindo criar *breakpoints* nos locais necessários.

3.4.7.4. Desenvolver o funcionamento

O seguinte excerto de código, ilustrado no Anexo 1 contém a seguinte lógica. Ao recolher o nó com os perfis de LinkedIn, o módulo percorre de forma recursiva cada nó filho, recolhendo os metadados pretendidos e, para cada perfil encontrado, cria um novo nó, com a informação estruturada de seguinte forma:

- Nó raiz – Primeiro e último nome (Ex: Vladyslav Adamovych)
  - Primeiro nó filho – Primeiro nome (Ex: Vladyslav)
  - Segundo nó filho – Último nome (Ex: Adamovych)
  - Terceiro nó filho – Endereço URL do perfil (Ex: https://pt.linkedin.com/...)
  - Quarto nó filho – Endereço URL do ícone do perfil (Ex: https://...)

3.4.7.5. Verificar o funcionamento

Para confirmar que o módulo foi reconhecido pela aplicação, deve ser possível poder encontrá-lo na lista dos módulos de processamento na opção "Process Module", premindo o botão direito num nó. Após execução do módulo, verificamos a criação de um novo nó com a informação extraída.

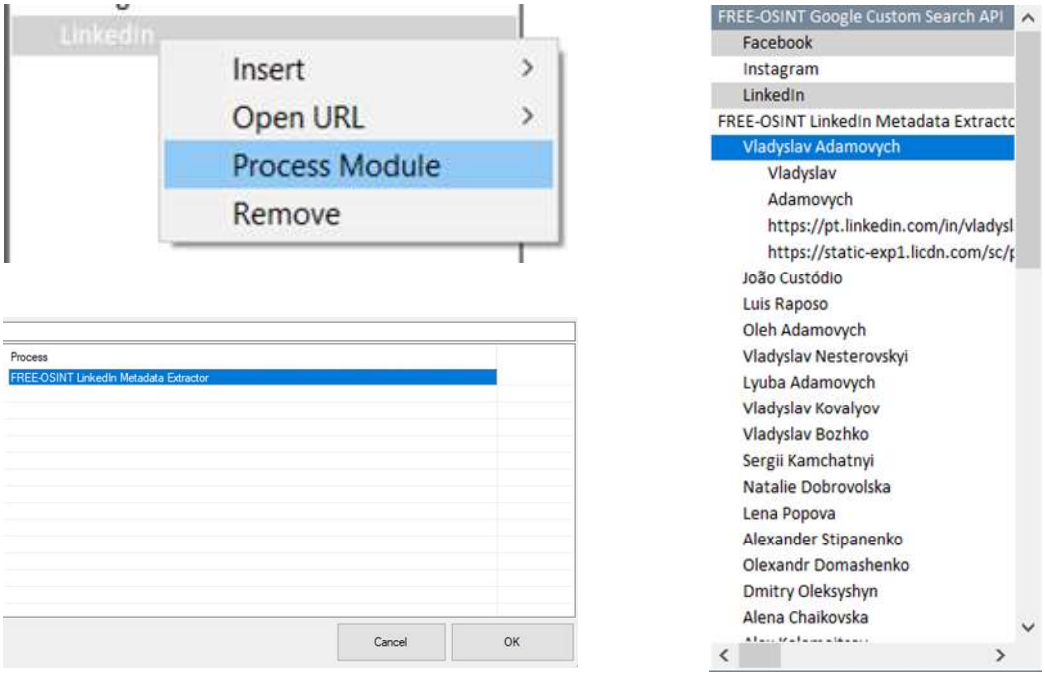


Figura 69 - Verificação do funcionamento

3.5.Sintese

Em resumo, com a base na pesquisa do mundo digital de OSINT foram identificadas inconsistências e potencial nas ferramentas atuais, o que permitiu estabelecer uma visão e desenvolver uma proposta. Em primeiro lugar, foram definidos os requisitos funcionais e não funcionais. Também se angariaram algumas inspirações de ferramentas populares e extraíram-se caraterísticas relevantes.

Isto resultou numa solução dividida em cinco projetos interligados. O primeiro projeto é a aplicação principal com estrutura modular, foco na interação intuitiva e capacidade de gestão da informação. Esta aplicação é complementada pelos restantes projetos em forma de módulos, com a exceção do NodeControl. Cada módulo desenvolvido reflete um exemplo de cada subprocesso do ciclo OSINT. Os objetivos dos módulos desenvolvidos também são de mostrar o potencial da estrutura modular da aplicação principal e oferecer algumas funcionalidades base.

O módulo FREE-OSINT Google Custom Search é um projeto que permite ao utilizador usufruir da Custom Search API da Google para a realização de consultas em motores de pesquisa personalizados. Este módulo partilha o foco da aplicação em relação a interação intuitiva, gestão da informação e flexibilidade através de vários mecanismos dedicados. O módulo FREE-OSINT LinkedIn_Metadata_Extractor foi desenvolvido como um exemplo que ilustrava os passos para criar um módulo simples da aplicação principal. Este, sendo um módulo de processamento, usufrui dos dados específicos, vindos do módulo FREE-OSINT Google Custom Search, e permite extrair informação a partir dos metadados dos resultados. O módulo FREE-OSINT Report_Builder foi desenvolvido para oferecer ao utilizador uma forma de documentar a pesquisa realizada através da aplicação principal, mantendo ao mesmo tempo a opção de modificação da estrutura do documento e dos seus dados. O projeto FREE-OSINT NodeControl é uma forte modificação de um projeto de fonte aberta existente, que permite ao utilizador interagir com os dados em forma de diagramas de contentores sincronizados com a área de trabalho.

Por fim, a solução atual oferece uma vasta variedade de funcionalidades. É permitido desenvolver funcionalidades acrescidas em forma de módulos que serão interpretadas e integrados no processo de aquisição. Isto com vários mecanismos de gestão de informação, foco na interface intuitiva e um conjunto de extra funcionalidades para melhorar a experiência do utilizador.

4. Demonstração e validação da solução

Após o desenvolvimento e implementação da aplicação proposta, pretende-se, neste capítulo, apresentar a demonstração de funcionamento e a validação da acomodação dos objetivos traçados inicialmente por parte da aplicação desenvolvida. Para tal, estas fases foram desenvolvidas através da realização de inquéritos e entrevistas feitos a utilizadores, assim como através da realização de casos de uso para demonstrar as funcionalidades da aplicação.

4.1. Casos de uso

A capacidade de realização de aquisição de OSINT exclusivamente através da aplicação será verificada através de um conjunto de casos de uso, tendo em conta algumas limitações. O estudo será realizado com a utilização exclusiva da aplicação principal em conjunto com seguintes módulos:

- FREE-OSINT Google Custom Search
- FREE-OSINT Report Builder

Tendo em consideração a limitação do módulo FREE-OSINT Google Custom Search quanto ao número diário de consultas, para o estudo será atribuído o limite de 1000 resultados a cada alvo. Será avaliada a quantidade e a qualidade da informação obtida. Realizar-se-ão dois casos de estudo com alvos diferentes. Para os efeitos do estudo os alvos serão entidades/figuras públicas, para evitar possíveis conflitos de privacidade no futuro. Os alvos escolhidos são os seguintes:

- Mestrado de Cibersegurança e Informação Forense (MCIF) do Instituto Politécnico de Leiria.
- Instituto Politécnico de Leiria (IPLeiria).

Para as consultas realizadas, no módulo Google Custom Search, foram utilizados seguintes parâmetros em combinações diferentes:

- Engine
  - LinkedIn – dedicada a pesquisas apenas no domínio de LinkedIn.
  - Facebook – dedicada a pesquisas apenas no domínio de Facebook.
  - Instagram – dedicada a pesquisas apenas no domínio de Instagram.
  - IPLeiria – dedicada a pesquisas no domínio de IPLeiria.
  - General – dedicada a pesquisas em toda a web.
- Documents From
  - Portugal
- File type
  - .pdf
- Geolocation
  - Portugal
- Document Language
  - Portuguese

Os resultados obtidos foram extraídos utilizando FREE-OSINT Report Builder e anexados no documento. Toda a informação é obtida exclusivamente pela aplicação.

4.1.1. Mestrado de Cibersegurança e Informação Forense

O seguinte caso de pesquisa envolve o Mestrado de Cibersegurança e Informação Forense do Instituto Politécnico de Leiria. O objetivo é encontrar o número máximo de endereços possível para o limite de consultas onde o MCIF se encontra referido. Os termos de consulta utilizados são seguintes:

- MCIF
- MCIF Leiria
- MCIF IPL
- MCIF IPLeiria
- MCIF docentes
- Mestrado de Cibersegurança e Informação Forense
- Mestrado de Cibersegurança e Informação Forense Leiria

Após análise e seleção dos resultados, foi criado um workspace com 23 endereços relevantes. Todos os endereços foram sub-categorizados em: Páginas do IPLeiria, Redes Sociais e Alunos, ilustrado pela Figura 70.

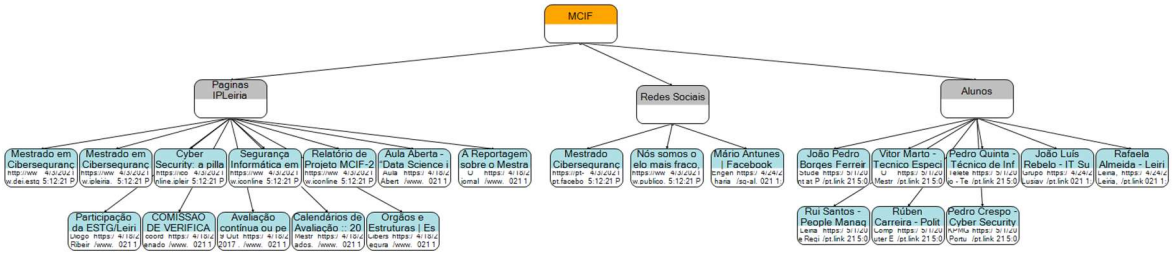


Figura 70 - Workspace de MCIF

Na categoria de páginas do IPLeiria encontra-se o endereço para a página principal do mestrado, no domínio do departamento de engenharia informática, e na diretoria de cursos do IPLeiria. Foi encontrado também o calendário de avaliação do 1º semestre de 2017/2018 do regime pós laboral, bem como a página com calendários de avaliação de todos os cursos para o ano 2018/2019. Para além disto, foi também recolhido um endereço onde estão identificadas as figuras representantes de vários departamentos, dos quais é identificado o coordenador do MCIF e o seu endereço de email. Os restantes endereços apontam para projetos/dissertações do mestrado.

Na categoria das redes sociais foi recolhido o endereço da página de MCIF no Facebook e uma publicação feita pelo coordenador do curso nesta. Também foi recolhida uma reportagem do "Público" feita sobre o alvo. A categoria dos alunos contém perfis de LinkedIn que identificam o mestrado alvo no seu percurso académico. No total foram recolhidos 23 resultados relevantes.

4.1.2. Instituto Politécnico de Leiria

O seguinte caso de pesquisa envolve o Instituto Politécnico de Leiria. O objetivo é encontrar o número máximo de endereços possível para o limite de consultas onde o IPLeiria se encontra referido. Os termos de consulta utilizados são seguintes:

- IPL
- IPL Leiria
- IPLeiria
- Instituto Politécnico de Leiria

Após análise e seleção dos resultados, foi criado um workspace com 322 endereços relevantes. Todos os endereços foram sub-categorizados em: General, PDF, Perfil LinkedIn e Instagram.

Na categoria “General” podemos encontrar todos os endereços relevantes que não podem ser categorizados pelas restantes categorias. A maioria dos resultados são subdomínios do IPLeiria com utilidade especifica. A categoria com o nome ‘PDF’ contém documentos do formato (.pdf) que fazem referência ao alvo. Muitos destes são projetos académicos. “Perfil LinkedIn” é a categoria que contém perfis de LinkedIn que, de alguma forma fazem referência ao alvo. Podemos encontrar nesta categoria: alunos, professores e funcionários. A categoria Instagram contém perfis e publicações que fazem referência ao alvo. No total, foram recolhidos 322 resultados relevantes e únicos, ilustrados pela Figura 71.

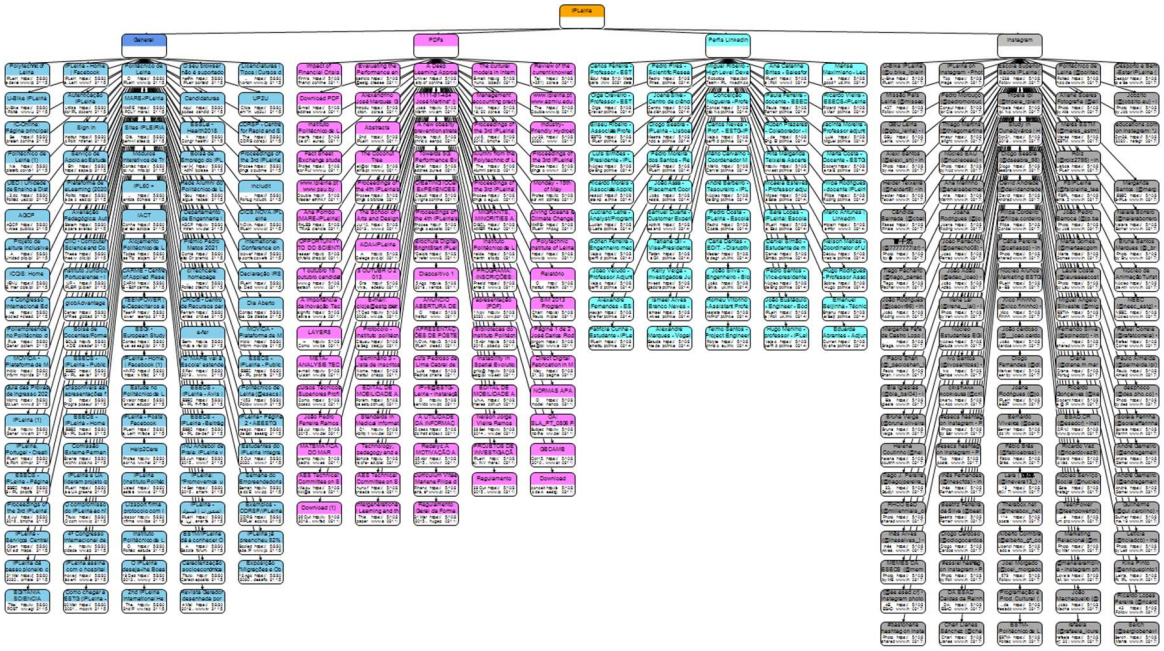


Figura 71 - Workspace de IPLeiria

4.2.Validação

Neste capítulo são analisadas as implementações efetuadas comparativamente aos objetivos e requisitos definidos. São apresentados os resultados da implementação relacionados com a usabilidade, estrutura e gestão da informação.

4.2.1. Usabilidade

Durante todo o processo de desenvolvimento, a usabilidade tem sido um grande fator na tomada de decisões. Qualquer funcionalidade desenvolvida tem de ser facilmente acessível e completamente compreensível. Os testes e o *feedback* recebidos acerca da usabilidade afetaram a aplicação de tal forma, que a versão atual perdeu toda a semelhança visual com a primeira versão funcional, ilustrados em Figura 72 e Figura 73.

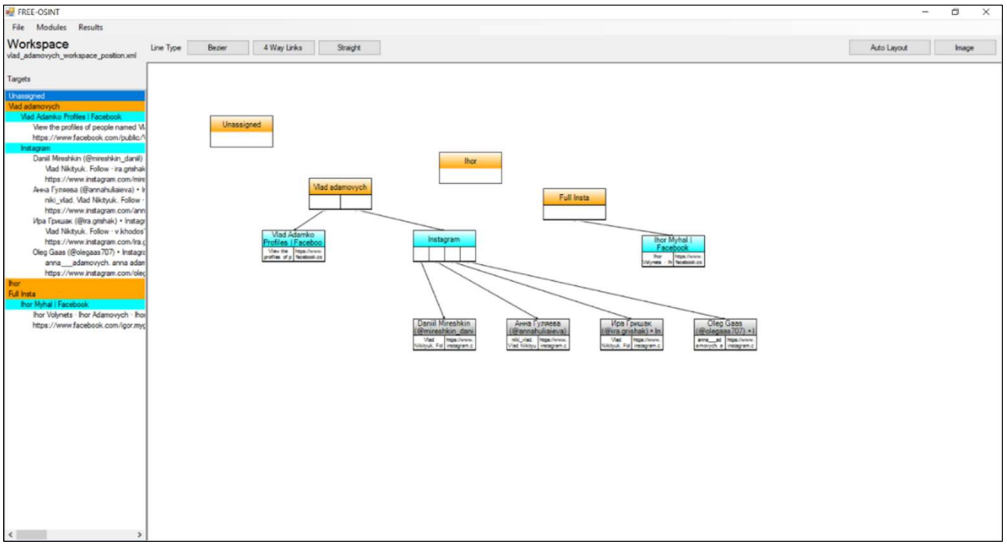


Figura 72 - Aplicação Principal (1ª fase)

Durante processo de desenvolvimento, no momento em que as funcionalidades pretendidas foram consideradas como desenvolvidas, partilharam-se várias cópias da ferramenta para um conjunto de pessoas com perfis profissionais diferentes. De seguida, estas foram entrevistadas de forma simples e informal, relativamente às suas experiências durante a utilização da ferramenta. No total, foram entrevistados oito indivíduos, dos quais, três são estudantes de Engenharia Informática, dois são licenciados em Engenharia Informática, que trabalham em diversas áreas, um é estudante de MCIF, o orientador do projeto e um inspetor policial.

Apesar do feedback ser geralmente positivo, os entrevistados tinham um conjunto de sugestões para a melhoria na interação. O *feedback* recolhido foi classificado pela ordem de importância e proximidade da área de OSINT e cibersegurança.

Através destas sugestões, a aplicação ganhou um conjunto de funcionalidades e caraterísticas novas. Dos exemplos mais notáveis temos os seguintes:

- Foi adicionado um navegador embutido e a possibilidade de abrir URL num predefinido no sistema operativo. Isto devido às afirmações dos utilizadores relacionadas com a dificuldades na verificação dos endereços, tendo que copiar manualmente o URL para um navegador.
- Também se redesenhou o painel do *workspace* para se apresentar como um controlador de separadores, onde poderiam ser adicionados separadores para abrir páginas web ou resultados de pesquisas. Como estes eram, previamente, janelas separadas, um *feedback* que foi apontado relacionava-se o excesso de janelas que eventualmente surgia ao utilizador.
- Um *feedback* focado diretamente no processo de aquisição de OSINT, vindo da experiência do inspetor, sugeria a criação de uma funcionalidade dedicada ao início de uma nova a partir de informações existentes. No momento, para tal, foi desenvolvida a funcionalidade de *compose query* que é a responsável pelo processo de reinício da pesquisa. As ações de pesquisa, interação com os resultados e documentação, sendo frequentemente utilizadas, passaram para a página principal como atalhos, permitindo reduzir a quantidade de cliques necessários para as mesmas.
- Alguns dos entrevistados também apontaram para a falta de nitidez que se verificou devido aos computadores de *high pixel density* utilizados. Este problema foi resolvido e encontra-se uma opção para esse fim na janela principal para acesso rápido. Os utilizadores também sugeriram a inclusão do controlo dos aspetos de personalização e alteração dos parâmetros do contentor, como a cor, tamanho e fonte de texto. Estes também foram implementados e encontram-se associados na janela principal.
- Algumas definições sugeridas relativas às cores predefinidas para os níveis hierárquicos, assim como, o limite de resultados por linha para cada ramo também foram implementadas, encontrando-se na janela de configuração.

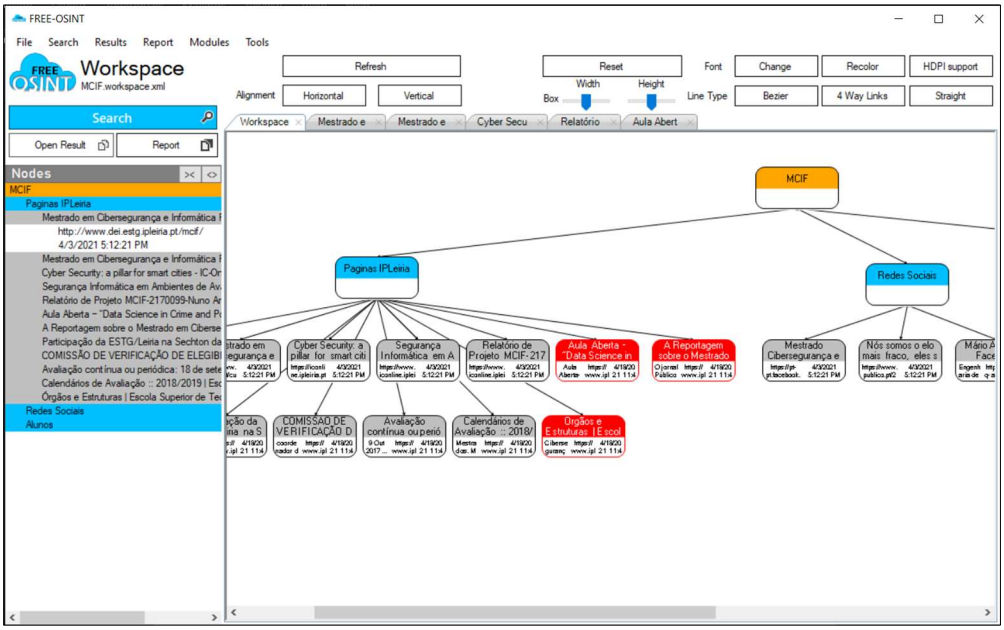


Figura 73 - Aplicação principal (2ª fase)

Um consenso entre os entrevistados estava relacionada com a dificuldade de compreensão do funcionamento geral durante a primeira utilização da aplicação. É um comportamento normal, pois os conceitos de *workspace*, resultados e o fluxo de trabalho geral são desconhecidos. No entanto, para remediar isto, foi publicado um vídeo com vários cenários de utilização, como a interação com os contentores no *workspace*, a configuração inicial do módulo FREE-OSINT Google Custom Search e um cenário de utilização que percorre o ciclo de aquisição OSINT. O vídeo está publicado no YouTube e tanto pode ser encontrado pesquisando pelo título da Figura 74 "FREE-OSINT Features and Examples", ou através do link na descrição do projeto no GitHub.

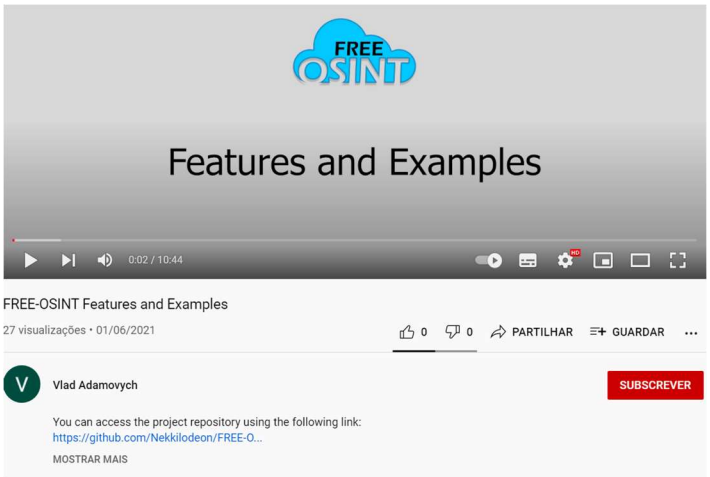


Figura 74 - FREE-OSINT Features and Examples

Para além destas implementações, entre outras, permitiram melhorar a interatividade com a aplicação, o que se confirmou pelo *feedback* resultante das segundas entrevistas recolhidas nos meses de agosto, setembro e outubro, pois dependeu da disponibilidade dos entrevistados. Tendo em conta que os entrevistados foram os mesmos, estes já tinham alguma familiaridade com a ferramenta, podendo ou não, confirmar o cumprimento das suas sugestões e fornecer opiniões acerca do funcionamento geral.

É importante destacar o *feedback* do entrevistado mais relevante. Visto que a aplicação se foca primeiramente na aquisição de OSINT, a opinião do inspetor tem maior importância e aqui se acaba por destacar de seguida.

Foram colocadas ao inspetor as seguintes questões:

Q: Considerou a aplicação Intuitiva? (Caso não considere, por favor indique do que sentiu falta)

R: “A configuração inicial e criação de "casos" pode criar alguma confusão ou configuração errada. Podia, na minha opinião, ser mais interativa com o utilizador (tipo, pergunta - resposta), ou colmatada com tutorial passo a passo após primeira utilização. No entanto, é apenas um detalhe, pois no geral a ferramenta é intuitiva e de fácil utilização”.

Q: O que sente ao nível de complexidade? (Se é difícil de utilizar e/ou se as ações fazem sentido)

R: “Atendendo à semelhança gráfica e organizacional com outras ferramentas forenses, a utilização por profissionais é fácil, apenas requer alguma habituação na configuração de módulos e critérios de pesquisa”.

Q: O que sente ao nível da gestão de informação? (Se o formato de grafo para os dados e o tipo de armazenamento fazem sentido)

R: “A gestão da informação recolhida quer de forma gráfica, quer na visualização de detalhe ou, mesmo na exportação de resultados (salvar os ficheiros, gerar relatório) está muito bem conseguido e é bastante útil”.

Q: Sentiu algumas limitações? (Caso sinta que faz falta alguma funcionalidade ou algo que lhe deixou uma experiência negativa)

R: “As únicas limitações foram apenas as que já referi antes, nas perguntas anteriores”.

Fora de questões referidas, o inspetor também apontou o seguinte comentário:

“Tenho de confessar que descarreguei a ferramenta e andei um pouco perdido, até conseguir recolher alguma informação”.

Este *feedback*, em conjunto com a sugestão de tutorial contribuíram para o desenvolvimento de janelas de *tips*, ou dicas. São janelas de pop-up, com *clips* de instruções e texto, que surgem em ocasiões específicas, para guiar o utilizador a utilizar ações necessárias durante o processo de aquisição, analise e configuração. Um exemplo desta janela pode ser visualizado na Figura 75.

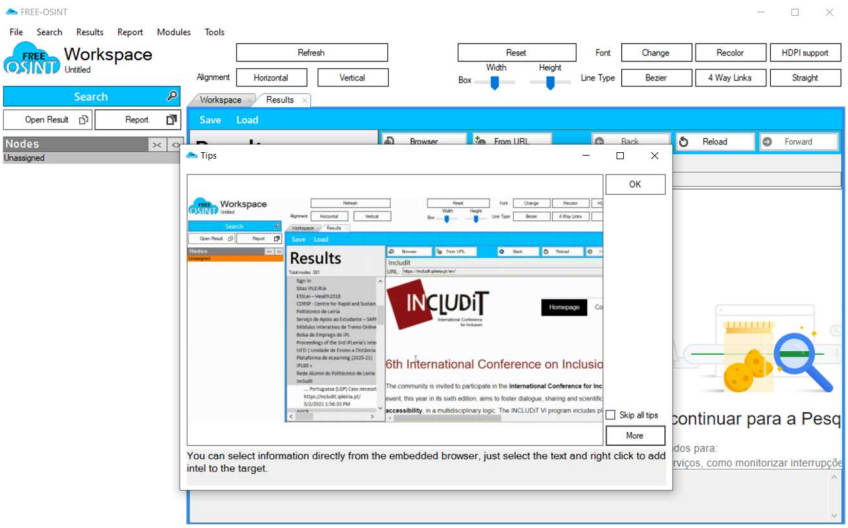


Figura 75 - Janela de Tips

Para terminar o tópico de feedbacks e sugestões, o inspetor sumarizou a sua experiência de seguinte forma:

“O tratamento gráfico da informação recolhida está ótimo (workspace), a capacidade organização visual dos ‘items’ em análise é extremamente intuitiva e de fácil manipulação, o acesso aos resultados e relatórios gerados é fácil e muito útil, o que aumenta a operacionalidade desta ferramenta para os profissionais, analistas ou investigadores. A semelhança gráfica às ferramentas mais usadas de Informática Forense é genial, na verdade é agradável para os profissionais do meio ter uma homogeneidade no uso das diversas ferramentas e isso foi muito bem conseguido”.

4.2.2. Estrutura Flexível e Adaptativa

A estrutura flexível e adaptativa foi alcançada através da modularidade da aplicação e separação dos processos de aquisição em diversas funcionalidades. O código que lê e interpreta aplicações executáveis (.exe) e bibliotecas (.dll) permite criar aplicações ou módulos dedicados a pesquisa, processamento e relato, ou adaptar ferramentas existentes e aplicá-los na aplicação FREE-OSINT.

Cada módulo de pesquisa desenvolvido ou adaptado para a estrutura FREE-OSINT tornará a solução mais resistente ao tempo e às mudanças nas fontes. Os módulos de processamento desenvolvidos ou adaptados irão fortalecer os de pesquisa, complementando as suas caraterísticas com funcionalidade extra. Os módulos de documentação permitirão transformar a informação dos *workspaces* em vários formatos ou até criar outras formas de interagir com esta, pois não existem restrições nos módulos, sendo apenas feita a passagem da informação. A falta de restrições cria muitas possibilidades, como por exemplo, um módulo de processamento pode até conter um ambiente próprio para interagir com a informação e devolver aquela que o utilizador escolheu.

Para além disto, todo o código encontra-se num repositório de fonte aberta e acessível a todos, permitindo criar todas as funcionalidades ou caraterísticas que forem necessárias sem quaisquer restrições.

4.2.3. Gestão da Informação

A gestão da informação em aplicações OSINT, torna-se útil não só em investigações de maiores dimensões como em pesquisas simples. Os *workspaces* e resultados foram concebidos para permitir ao utilizador interagir com a informação obtida a partir da aplicação, sendo entidades distintas por este motivo. Tendo em conta que, durante o processo da recolha dos dados, a quantidade de informação obtida é potencialmente enorme, o que prejudica a interatividade com um *workspace*, seria necessário desenhar um conjunto enorme de contentores e sincronizá-los com a TreeView. Esse facto, para além de prejudicar a *performance* da aplicação, irá criar confusão visual desnecessária. A janela de resultados é ideal para este tipo de interação, pois tem a informação de todos os módulos numa lista com níveis e um navegador embutido para acesso rápido à informação. Caso a informação obtida seja relevante, o investigador inclui o nó que a contém no *workspace*.

Para além disto, toda esta informação pode ser armazenada, o que possibilita a retoma da investigação a partir do ponto de armazenamento e também partilhar as informações com colegas ou outros dispositivos de trabalho.

Esta funcionalidade é alcançada de várias formas. Atualmente, a aplicação principal lê e interpreta três tipos de informação a partir de ficheiros XML.

Os resultados de recolha da informação ou pesquisa, que são devolvidos em forma de TreeView, podem ser armazenados ou importados a partir de ficheiros. Isto remove a necessidade de realização de pesquisas adicionais para aceder ao mesmo *dataset* de resultados. Com esta abordagem poderá ser atribuído ao conjunto de módulos de pesquisa a função de agregar o máximo de informação possível, armazená-la num ficheiro e distribuir por vários colegas para a análise e seleção coletiva dos dados. As workspace seguem a mesma lógica, pois são importados e extraídos com todos os dados que lhes são associados como ligações entre nós, bem como as posições e cores destes. O ficheiro de configuração, como o nome indica, permite manter as configurações definidas para as interações futuras.

A alteração do sufixo para cada tipo de ficheiro permite utilizar a mesma diretoria, tanto para armazenar *workspaces* como resultados e não criar confusão, pois a janela de importação irá tratar da filtragem e apresentar apenas o que é relevante para o contexto, o que é exemplificado na Figura 76.

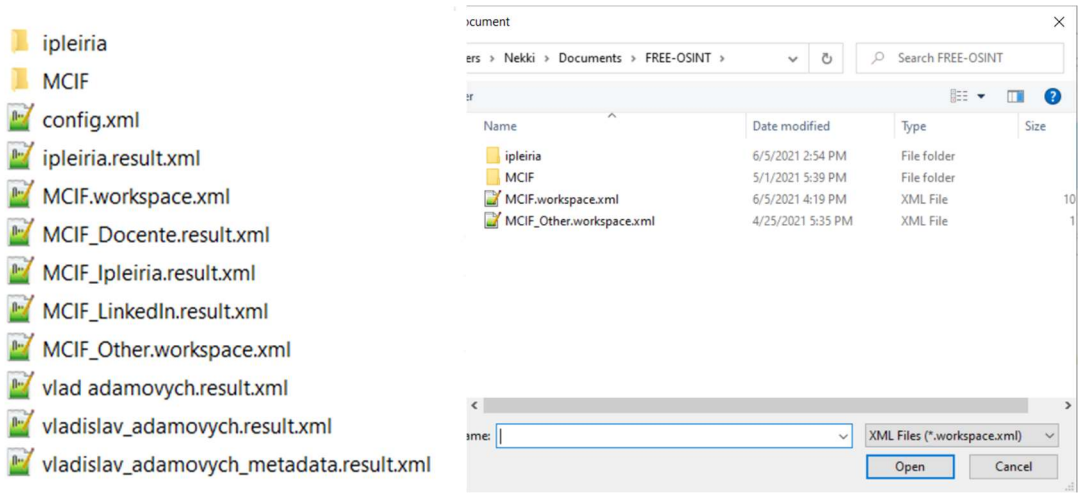


Figura 76 - Janela de importação do *workspace*

4.3.Síntese

Em resumo, para verificar se a solução cumpre os objetivos definidos foi realizada uma análise. É realizada uma comparação entre as características da implementação e os objetivos de usabilidade, estrutura e gestão da informação.

Para a usabilidade definiram-se momentos de avaliação e *feedback* com utilizadores onde, o *feedback* servia de fundamento para a continuação do desenvolvimento. A última fase de avaliação permitiu consolidar a perspetiva de um utilizador comum, devido à unanimidade na opinião dos entrevistados.

O cumprimento do objetivo que é a estrutura flexível é justificado pela modularidade da aplicação e separação das partes de cada subprocesso do ciclo de OSINT em funcionalidades distintas que, respeitam a volatilidade do processo.

O objetivo de gestão da informação é comparado aos mecanismos de gestão e armazenamento dos *workspaces*, resultados de pesquisa e configurações com uma variedade de parâmetros relativos a cada um. Para alem disso, a solução também conta com um conjunto de mecanismos de *quality of life* para a gestão oferecida como, o exemplo dos resultados e *workspaces* serem automaticamente filtrados na janela de armazenamento.

Por fim, é demonstrado um par de casos de uso onde, se verificam os possíveis resultados tanto em forma de diagramas de contentores como, documentos de relatórios construídos através da solução.

5. Conclusão

Como foi referido anteriormente, o ambiente de OSINT está em constante mudança, da mesma forma que as fontes abertas estão em constante mudança. As tentativas da RGPD acompanhar e regularizar os novos serviços, inviabiliza várias ferramentas de aquisição de OSINT que podem já existir para estes serviços. No entanto, o processo de aquisição de OSINT não mudou. Mudaram apenas os meios de recolha dos dados e os requisitos criados para a aplicação refletem esta realidade.

A pesquisa exploratória e comparativa das fontes, ferramentas e ambientes de OSINT permitiu melhorar a perspetiva acerca das dificuldades atuais e simultaneamente identificar aspetos promissores. Estes aspetos consequentemente, levaram à definição de um conjunto de critérios que uma ferramenta de OSINT deverá respeitar, que são a usabilidade, estrutura flexível e gestão da informação. Através de uma estrutura flexível, a solução será mais resiliente à volatilidade do mundo de OSINT e permitirá um desenvolvimento e melhoria contínuos. Uma interface intuitiva será mais acessível a todo o tipo de utilizador e permitirá focar na investigação imediatamente. Os mecanismos de gestão da informação oferecerão a possibilidade de organizar e armazenar os dados da investigação.

Com estes conceitos foi desenvolvida uma solução designada por FREE-OSINT, um projeto de fonte aberta que é composto por vários projetos. A biblioteca FREE_OSINT_LIB é responsável pela modularidade da aplicação principal e a organização dos seus módulos. O projeto FREE-OSINT Node Control oferece um ambiente de interação ao utilizador em forma de diagrama de contentores sincronizados com a área de trabalho. A biblioteca CefSharp permite ao utilizador aceder à informação diretamente na aplicação e extraí-la para o ambiente de trabalho através de um navegador embutido. O projeto FREE-OSINT Google Custom Search é implementado como um módulo que permite ao utilizador realizar pesquisas através da API da Google e incluir os resultados na área de trabalho. O projeto FREE-OSINT LinkedIn Metadata Extractor é implementado como um módulo que usufrui do módulo referido anteriormente para transformar os resultados num conjunto de dados mais organizados através da extração dos seus metadados. E por último, mas não menos importante, o projeto FREE-OSINT Report Builder, que é implementado como um módulo que permite a construção de relatórios através da integração da ferramenta Microsoft Word, para possibilitar a posterior personalização e exportação para outros formatos.

Todo este conjunto de projetos constitui a solução de FREE-OSINT e a sua capacidade de corresponder aos objetivos definidos é verificada através de uma interpretação dos seus mecanismos relativamente a cada objetivo definido.

A estrutura flexível é obtida através da arquitetura modular, pois permite aos utilizadores desenvolver funcionalidades extra que, por sua vez, irão assegurar a relevância desta com a passagem do tempo. Sempre que surge uma nova forma de partilhar os dados na rede pública, poderá ser desenvolvido um módulo que usufrua das suas funcionalidades. Para além disto, a abstração da informação recolhida e extraída permite expandir a área de utilizadores alvo, como por exemplo, um módulo que consome a API de um serviço de aquisição de vulnerabilidades de um domínio, permite às organizações recolher informações sobre alguns mecanismos de segurança que estes têm aplicados, transformando a aplicação numa ferramenta de *footprinting* para domínios.

A capacidade de gerir a informação é justificada pela estrutura e os mecanismos de gestão e organização de *workspaces*, resultados e semelhantes. Os *workspaces* representam o ambiente de trabalho em forma de *datasets* com informações acerca dos contentores, as suas posições, cores e hierarquia. Os resultados representam também de forma hierárquica a saída dos módulos de pesquisa e são transformados em itens no *workspace* através das decisões dos utilizadores. Estes, em conjunto com as configurações, são geridos e armazenados pela aplicação principal e os seus módulos.

Para validar o objetivo da interação intuitiva foi realizada uma distribuição de cópias da solução em dois momentos distintos do desenvolvimento e a posterior entrevista informal. O *feedback* obtido durante a primeira distribuição resultou no desenvolvimento de funcionalidades para melhorar a interação. O resultado deste desenvolvimento foi verificado durante a segunda distribuição e recolha de *feedback*. A unanimidade no *feedback* positivo permitiu concluir que o objetivo de interação intuitiva foi concretizado.

Com isto é possível afirmar que a concretização de todos os objetivos permitiu oferecer uma boa base para desenvolvimento e melhoria continua, com o máximo de funcionalidades possível para acelerar o processo. É esperado que a promessa da solução e as funcionalidades existentes demostrem o seu potencial e desperte o interesse da comunidade.

5.1.Limitações

No momento a aplicação conta com apenas um meio de aquisição da informação (Google Custom Search) e apenas uma forma de construção de relatórios, o módulo que usufrui da ferramenta Microsoft Word.

Durante o processo de utilização e teste da aplicação, um problema recorrente e relativamente comum para o cenário de uso, é o facto de existir a impressão digital do navegador. Este tipo de informação partilhada pelo navegador embutido previne a usufruição total das suas capacidades em serviços web, com mecanismos contra o acesso automático aos dados. Por exemplo, na tentativa de análise de vários perfis no LinkedIn, após uma certa quantidade de acessos, é restrita a visualização destes sem realizar antes o login ou registo.

É de notar que o sucesso da aplicação depende das comunidades de inteligência e cibersegurança. O código é de fonte aberta para a aplicação e os seus módulos e encontram-se disponíveis ao público num repositório GitHub, limitando-se apenas aos licenciamentos BSD e MIT de bibliotecas nele usadas.

5.2.Contribuições

Tendo em conta os objetivos estabelecidos e os resultados alcançados, são agora apresentadas as principais contribuições desta investigação.

- Levantamento das questões relacionadas com as formas de aquisição de OSINT no atual ambiente digital, os seus aspetos positivos e as dificuldades.

A primeira das contribuições consistiu no levantamento das questões relacionadas com as formas de aquisição de OSINT atuais. Para este efeito foi realizada uma pesquisa exploratória das ferramentas, técnicas, ambientes e fontes atuais. Pretendeu-se identificar desafios no processo de aquisição face às formas de obtenção de informação OSINT populares. Isto permitiu identificar problemas como: a dominância de ferramentas em linhas de comando, e(ou) dedicados apenas a um serviço ou fonte, a constante mudança dos meios de obtenção da informação e as respetivas fontes e a falta de opções de fonte aberta que ofereçam caraterísticas necessárias para completar um ciclo de OSINT.

- Proposta de uma solução que permita completar um ou mais ciclos de OSINT com caraterísticas de ferramenta acessível e uma estrutura apropriada para o processo.

Na sequência da análise anterior foram definidos um conjunto de requisitos e caraterísticas que uma ferramenta completa de OSINT deverá ter. Estes requisitos e caraterísticas foram então transformados numa arquitetura concreta. Esta dividiu-se em conceitos para mecanismos de interação, modularidade e gestão da informação.

- Desenvolvimento da proposta de solução utilizando a arquitetura definida.

As contribuições anteriores resultaram na necessidade de validação, no que se refere à capacidade da arquitetura definida numa aplicação real de oferecer uma contribuição legitima para a comunidade de inteligência e cibersegurança. Para este efeito, foi implementado um conjunto de projetos de fonte aberta que constituíram a solução final. Solução esta foi então validada através da interpretação dos seus mecanismos relativamente às dificuldades das ferramentas atuais e os objetivos definidos bem como, a distribuição e recolha de *feedback* por pessoas que são, ou não, relacionadas com a área. A solução está separada em cinco repositórios publicados individualmente. Os repositórios são descritos da seguinte forma:

  - FREE-OSINT – repositório que contém a aplicação principal em conjunto com a biblioteca partilhada FREE-OSINT-Lib, pois estes estão interligados e as alterações realizadas na *lib*, terão de ser interpretados pela aplicação principal. Este repositório está localizado no seguinte endereço: github.com/Nekkilodeon/FREE-OSINT
  - FREE-OSINT_Node_Control – repositório que contém a versão modificada da biblioteca node control. Este repositório está localizado no seguinte endereço: github.com/Nekkilodeon/FREE-OSINT_Node_Control
  - FREE-OSINT_Report_Builder – repositório que contém o módulo de relato que gera um documento Word a partir de nós selecionados. Este repositório está localizado no seguinte endereço: github.com/Nekkilodeon/FREE-OSINT_Report_Builder
  - FREE-OSINT_Google-Custom-Search – repositório que contém o módulo de pesquisa que permite interagir com a API da Google Custom Search com o propósito de aquisição de informação OSINT. Este repositório está localizado no seguinte endereço: github.com/Nekkilodeon/FREE-OSINT_Google-Custom-Search

- FREE-OSINT_LinkedIn_Metadata_Extractor – repositório que contém o módulo de processamento desenvolvido para explicar a construção de um módulo. O módulo permite recolher a partir dos resultados de LinkedIn com metadados a informação do primeiro e último nome, o endereço e o ícone destes. Este repositório está localizado no seguinte endereço: github.com/Nekkilodeon/FREE-OSINT_LinkedIn_Metadata_Extractor

- Comunicação

Após validação da solução, decidiu-se desenvolver a comunicação através do envio de uma publicação científica.

- Adamovych, V. e Santos L. (2021, November). FREE-OSINT. In *European Conference on Cyber Warfare and Security*. Academic Conferences International Limited.

5.3. Trabalho futuro

Embora a solução atual ofereça um conjunto vasto de funcionalidades, ainda existem formas de aumentar o seu potencial.

O desenvolvimento da funcionalidade que permite executar *scripts* codificados com linguagens diferentes e transformar o seu input em TreeNodes permitirá usufruir de aplicações já existentes sem a necessidade de as adaptar, como por exemplo módulos existentes em Recon-ng ou até a ferramenta em si.

Para além disto, o projeto NodeControl de FREE-OSINT poderá beneficiar bastante com desenvolvimento de outras formas de auto alinhamento ou organização dos contentores para permitir maior claridade visual na interação com os dados.

Desenvolver um histórico de ações realizadas no *workspace*, para que seja possível retroceder nos eventos ou em modificações indesejadas nos contentores, como por exemplo utilizando o atalho CTRL-Z.

Outro desenvolvimento que beneficiará bastante o desempenho na interação e manipulação da informação em *workspaces* da aplicação principal, será a utilização de múltiplos *threads* no desenho dos contentores.

Uma possível solução a estudar para a limitação do navegador, referido no subcapítulo de limitações, será o *spoofing* da impressão digital do navegador e(ou) do endereço IP.

Por último, e o mais importante, não pode ser ignorada a necessidade de desenvolvimento de novos módulos para expandir o potencial da solução. É necessário reiterar que, para a solução desenvolvida ter o seu impacto e ajudar na promoção de um ciberespaço mais seguro é crucial a contribuição da comunidade.

Bibliografia ou Referências Bibliográficas

[1] "Intelligence Studies: Home," [Online]. Available: https://usnwc.libguides.com/intelligence. [Acedido em 30 07 2021].

[2] "Nato Open Source Intelligence Handbook," 2001. [Online]. Available: https://www.academia.edu/4037348/NATO_Open_Source_Intelligence_Handbook. [Acedido em 21 Outubro 2020].

[3] K. P. K. Yogish Pai U, "Open Source Intelligence and its Applications in Next Generation Cyber Security - A Literature Review," *SRINIVASPUBLICATION,* vol. 5, nº 2, pp. 3-4, Agosto 2021.

[4] R. Chan, "The Cambridge Analytica whistleblower explains how the firm used Facebook data to sway elections," 5 Otubro 2019. [Online]. Available: https://www.businessinsider.com/cambridge-analytica-whistleblower-christopher-wylie-facebook-data-2019-10. [Acedido em 21 Julho 2021].

[5] T. Magee, "Huawei controversies timeline," 10 Setembro 2019. [Online]. Available: https://web.archive.org/web/20200318092537/https://www.computerworld.com/article/3427998/huawei-controversies-timeline.html. [Acedido em 21 Julho 2021].

[6] T. Warren, "Zoom faces a privacy and security backlash as it surges in popularity," 1 Abril 2020. [Online]. Available: https://www.theverge.com/2020/4/1/21202584/zoom-security-privacy-issues-video-conferencing-software-coronavirus-demand-response. [Acedido em 21 Julho 2021].

[7] "Data privacy: What the consumer really thinks," Fevereiro 2018. [Online]. Available: https://dma.org.uk/uploads/misc/5a857c4fdf846-data-privacy---what-the-consumer-really-thinks-final_5a857c4fdf799.pdf. [Acedido em 21 Julho 2021].

[8] T. M.Paulson, Intelligence Issues and Developments, New York: Nova Science Publishers, 2008.

[9] C. Andrew, R. J. Aldrich e W. K. Wark, Secret Intelligence A Reader, New York: Routledge, 2009.

[10] A.-J. B. Alalawi N, "Social Network and Privacy," 2016. [Online]. Available: https://www.hilarispublisher.com/open-access/social-network-and-privacy-2165-7912-1000288.pdf. [Acedido em 20 Outubro 2020].

[11] “MySpace – História e evolução da rede social sucesso dos anos 2000,” 29 06 2020. [Online]. Available: https://segredosdomundo.r7.com/myspace-historia/. [Acedido em 19 Outubro 2021].

[12] “MySpace: The OSINT Left Behind For Collection,” 10 Jan 2019. [Online]. Available: https://www.forbes.com/sites/joegray/2019/01/10/myspace-the-osint-left-behind-for-collection/?sh=541d171f20fa. [Acedido em 19 Outubro 2020].

[13] “List of Operating Systems for OSINT (Open-Source Intelligence),” 5 Agosto 2020. [Online]. Available: https://pentestit.com/operating-systems-open-source-intelligence-osint-list/. [Acedido em 14 Fevereiro 2021].

[14] B. R. &. M. Frade, “Slides de aula,” [Online]. Available: ipleiria.pt. [Acedido em 14 Janeiro 2021].

[15] “Use Maltego to Fingerprint an Entire Network Using Only a Domain Name,” 31 05 2018. [Online]. Available: https://null-byte.wonderhowto.com/how-to/use-maltego-fingerprint-entire-network-using-only-domain-name-0184900/. [Acedido em 6 2 2021].

[16] S. Mittal, S. Chauhan e K. Aggarwal, “datasploit,” 24 Setembro 2020. [Online]. Available: https://github.com/DataSploit/datasploit.

[17] “tinfoleak,” [Online]. Available: http://tinfoleak.com.

[18] “SpiderFoot,” [Online]. Available: https://www.spiderfoot.net.

[19] “Metagoofil,” [Online]. Available: https://github.com/laramies/metagoofil.

[20] “Intrigue.io,” [Online]. Available: https://intrigue.io.

[21] “Recon-ng,” [Online]. Available: https://github.com/lanmaster53/recon-ng.

[22] “Gitrob,” [Online]. Available: https://github.com/michenriksen/gitrob.

[23] “Creating a personal access token,” [Online]. Available: https://docs.github.com/en/github/authenticating-to-github/keeping-your-account-and-data-secure/creating-a-personal-access-token. [Acedido em 31 Outubro 2020].

[24] “Operative Framework,” [Online]. Available: https://github.com/graniet/operative-framework.

[25] “The Harvester,” [Online]. Available: https://github.com/laramies/theHarvester.

[26] “BackTrack Linux,” [Online]. Available: https://www.backtrack-linux.org. [Acedido em 7 Julho 2021].

[27] “Artículo: Use the Buscador OSINT VM for Conducting Online Investigations,” [Online]. Available: https://www.ciberforense.com.es/?p=2755. [Acedido em 21 Julho 2021].

[28] “Xfce Desktop Environment,” [Online]. Available: https://xfce.org. [Acedido em 21 Julho 2021].

[29] “HURON,” [Online]. Available: https://github.com/HuronOsint/OsintDistro. [Acedido em 21 Julho 2021].

[30] “Trace Labs Kali Linux,” [Online]. Available: https://github.com/tracelabs/tlosint-live. [Acedido em 22 Julho 2021].

[31] “Search Engine Definition,” [Online]. Available: https://techterms.com/definition/searchengine. [Acedido em 24 Setembro 2020].

[32] “Search Engine Market Share Europe,” 1 August 2020. [Online]. Available: https://gs.statcounter.com/search-engine-market-share/all/europe. [Acedido em 24 Setembro 2020].

[33] S. Brin e L. Page, “The Anatomy of a Large-Scale Hypertextual,” [Online]. Available: http://infolab.stanford.edu/pub/papers/google.pdf. [Acedido em 26 Setembro 2020].

[34] S. Levy, “Exclusive: How Google’s Algorithm Rules the Web,” 22 Fevereiro 2010. [Online]. Available: https://web.archive.org/web/20110416062117/http://www.wired.com/magazine/2010/02/ff_google_algorithm/. [Acedido em 26 Setembro 2020].

[35] “What Google 'Hummingbird' Means for Your SEO Strategy,” 15 Novembro 2013. [Online]. Available: https://www.entrepreneur.com/article/229926. [Acedido em 26 Setembro 2020].

[36] V. Beal, “SEO Meaning & Definition,” [Online]. Available: https://www.webopedia.com/TERM/S/SEO.html. [Acedido em 27 Setembro 2020].

[37] “Diretrizes para webmasters,” [Online]. Available: https://support.google.com/webmasters/answer/35769. [Acedido em 2020 Setembro 27].

[38] J. Hardwick, “Google Search Operators: The Complete List (42 Advanced Operators),” 24 Dezembro 2019. [Online]. Available: https://ahrefs.com/blog/google-advanced-search-operators/. [Acedido em 28 Maio 2020].

[39] “Custom Search JSON API | Programmable Search Engine,” [Online]. Available: https://developers.google.com/custom-search/v1/overview. [Acedido em 30 Setembro 2020].

[40] Justone, “Scraping Google for Fun and Profit,” [Online]. Available: http://google-scraper.squabbel.com. [Acedido em 27 Setembro 2020].

[41] G. Panagopoulos e C. Faloutsos, “Bit-Sliced Signature Files for Very Large Text Databases on a Parallel Machine Architecture”.

[42] “Microsoft Open Sources Major Components Of Bing Search Engine, Here's Why It Matters,” [Online]. Available: https://fossbytes.com/microsoft-working-open-source-search-components-used-power-bing/. [Acedido em 17 Outubro 2020].

[43] T. A. Usmani, D. Pant e A. K. Bhatt, “A Comparative Study of Google and Bing Search Engines in Context of Precision and Relative Recall Parameter,” *International Journal on Computer Science and Engineering (IJCSE),* vol. 4, nº 1, 2012.

[44] “Bing Places for Business,” [Online]. Available: https://www.bingplaces.com. [Acedido em 18 Outubro 2020].

[45] “Cognitive Services—APIs for AI Developers | Microsoft Azure,” [Online]. Available: https://azure.microsoft.com/en-us/services/cognitive-services/#features. [Acedido em 19 Outubro 2020].

[46] “Search Engine Market Share Russian Federation | StatCounter Global Stats,” [Online]. Available: https://gs.statcounter.com/search-engine-market-share/all/russian-federation. [Acedido em Outubro 29 2020].

[47] “Компания Яндекс — Технологии — Архитектура ответа на вопрос,” [Online]. Available: https://yandex.ru/company/technologies/searcharch. [Acedido em 1 Novembro 2020].

[48] “Query language - Mail. Help (Yandex),” [Online]. Available: https://yandex.com/support/mail/web/letter/query-language.html. [Acedido em 1 Novembro 2020].

[49] [Online]. Available: https://yandex.ru/dev/xml/doc/dg/concepts/captcha.html/. [Acedido em 1 Novembro 2020].

[50] “DuckDuckGo Privacy,” [Online]. Available: https://duckduckgo.com/privacy. [Acedido em 16 Novembro 2020].

[51] “Sources | DuckDuckGo Help Pages,” [Online]. Available: https://help.duckduckgo.com/results/sources/. [Acedido em 16 Novembro 2020].

[52] “Instant Answers and Other Features | DuckDuckGo Help Pages,” [Online]. Available: https://help.duckduckgo.com/duckduckgo-help-pages/features/instant-answers-and-other-features/. [Acedido em 17 Novembro 2020].

[53] “Instant Answers,” [Online]. Available: https://duck.co/ia. [Acedido em 17 Novembro 2020].

[54] “DuckDuckGo !Bang,” [Online]. Available: https://duckduckgo.com/bang?. [Acedido em 17 Novembro 2020].

[55] “How Google, Yahoo, and Bing Differ | WebFX,” [Online]. Available: https://www.webfx.com/internet-marketing/what-are-the-differences-among-google-yahoo-and-bing.html. [Acedido em 17 Novembro 2020].

[56] “Yahoo Search API | ProgrammableWeb,” [Online]. Available: https://www.programmableweb.com/api/yahoo-search. [Acedido em 17 Novembro 2020].

[57] “SOCIAL NETWORK | meaning in the Cambridge English Dictionary,” [Online]. Available: https://dictionary.cambridge.org/dictionary/english/social-network. [Acedido em 11 Outubro 2020].

[58] “Social Media Stats Europe | StatCounter Global Stats,” [Online]. Available: https://gs.statcounter.com/social-media-stats. [Acedido em 11 Outubro 2020].

[59] “Social Media Stats Europe | StatCounter Global Stats,” [Online]. Available: https://gs.statcounter.com/social-media-stats/all/europe. [Acedido em 11 Outubro 2020].

[60] “Most used social media platform | Statista,” Julho 2020. [Online]. Available: https://www.statista.com/statistics/272014/global-social-networks-ranked-by-number-of-users/. [Acedido em 2 Outubro 2020].

[61] N. Confessore, “Cambridge Analytica and Facebook: The Scandal and the Fallout So Far,” 04 Abril 2018. [Online]. Available: https://www.nytimes.com/2018/04/04/us/politics/cambridge-analytica-scandal-fallout.html. [Acedido em 4 Outubro 2020].

[62] “OSINT: Where were you when the Facebookalypse struck?,” 11 Junho 2019. [Online]. Available: https://medium.com/@osintessentials/osint-where-were-you-when-the-facebookalypse-struck-43b73ff0dd05. [Acedido em 4 Outubro 2020].

[63] “Pages Search,” [Online]. Available: https://developers.facebook.com/docs/pages/searching/. [Acedido em 11 Outubro 2020].

[64] “The 3 Facebook search workarounds everyone is talking about.,” [Online]. Available: https://www.paliscope.com/2019/07/09/the-3-facebook-search-workarounds-everyone-is-talking-about/. [Acedido em 12 Outubro 2020].

[65] “Pinterest: monthly active users worldwide 2020 | Statista,” [Online]. Available: https://www.statista.com/statistics/463353/pinterest-global-mau/. [Acedido em 13 Outubro 2020].

[66] “OSINT investigation on Pinterest - Aware Online Academy,” [Online]. Available: https://www.aware-online.com/en/osint-investigation-on-pinterest/. [Acedido em 20 Outubro 2020].

[67] “Twitter: monthly active users worldwide | Statista,” [Online]. Available: https://www.statista.com/statistics/282087/number-of-monthly-active-twitter-users/. [Acedido em 14 Outubro 2020].

[68] “Using Twitter,” [Online]. Available: https://help.twitter.com/en/using-twitter. [Acedido em 14 Outubro 2020].

[69] “User object | Docs | Twitter,” [Online]. Available: https://developer.twitter.com/en/docs/twitter-api/data-dictionary/object-model/user. [Acedido em 14 Outubro 2020].

[70] “Explore a user’s Tweets | Docs | Twitter,” [Online]. Available: https://developer.twitter.com/en/docs/tutorials/explore-a-users-tweets. [Acedido em 14 Outubro 2020].

[71] “Advanced Search,” [Online]. Available: https://twitter.com/search-advanced. [Acedido em 14 Outubro 2020].

[72] “How to Run Social Media Investigations,” [Online]. Available: https://traversals.com/blog/social-media-investigations/. [Acedido em 14 Outubro 2020].

[73] “Instagram: active users worldwide | Statista,” [Online]. Available: https://www.statista.com/statistics/253577/number-of-monthly-active-instagram-users/. [Acedido em 14 Outubro 2020].

[74] “What is Instagram?,” [Online]. Available: https://help.instagram.com/424737657584573. [Acedido em 14 Outubro 2020].

[75] “Instagram Developer Documentation,” [Online]. Available: https://www.instagram.com/developer/. [Acedido em 20 Outubro 2020].

[76] “YouTube by the Numbers (2020): Stats, Demographics & Fun Facts,” [Online]. Available: https://www.omnicoreagency.com/youtube-statistics/. [Acedido em 15 Outubro 2020].

[77] “Search: list | YouTube Data API | Google Developers,” [Online]. Available: https://developers.google.com/youtube/v3/docs/search/list. [Acedido em 30 Outubro 2020].

[78] “YouTube Data API Overview,” [Online]. Available: https://developers.google.com/youtube/v3/getting-started#quota. [Acedido em 30 Outubro 2020].

[79] [Online]. Available: https://www.statista.com/statistics/261925/unique-visitors-to-tumblrcom/. [Acedido em 30 Outubro 2020].

[80] “What is Tumblr and how is it used? [2018 Definition] BigCommerce,” [Online]. Available: https://www.bigcommerce.com/ecommerce-answers/what-tumblr-and-how-it-used/. [Acedido em 30 Outubro 2020].

[81] A. Waddell, “Guide to Open Source Intel Search Methods,” 1 Novembro 2017. [Online]. Available: https://www.researchgate.net/publication/320871796_Guide_to_Open_Source_Intel_Search_Methods.

[82] “Cross platform, open source .NET framework,” [Online]. Available: http://www.mono-project.com. [Acedido em 13 5 2021].

[83] “What Is C# Language, Advantages & Features Of C# Language,” 30 08 2019. [Online]. Available: https://www.codexoxo.com/advantages-c-sharp-language/. [Acedido em 20 Dezembro 2020].

[84] “drake7707/NodeControl,” 20 Abril 2017. [Online]. Available: https://github.com/drake7707/nodecontrol. [Acedido em 20 Outubro 2020].

[85] “Photoshop Layers Panel Essentials,” [Online]. Available: https://www.photoshopessentials.com/basics/layers/layers-panel/. [Acedido em 19 Fevereiro 2020].

[86] “cefsharp/CefSharp: .NET (WPF and Windows Forms) bindings for the Chromium Embedded Framework,” 9 Fevereiro 2021. [Online]. Available: https://github.com/cefsharp/CefSharp. [Acedido em 18 Fevereiro 2021].

[87] “Licença de SefSharp,” [Online]. Available: https://github.com/cefsharp/CefSharp/blob/master/LICENSE. [Acedido em 13 Maio 2021].

[88] “FREE-OSINT Node Control,” [Online]. Available: https://github.com/Nekkilodeon/FREE-OSINT_Node_Control. [Acedido em 14 Maio 2021].

[89] “Method: cse.list | Custom Search JSON API | Google Developers,” [Online]. Available: https://developers.google.com/custom-search/v1/reference/rest/v1/cse/list#request. [Acedido em 20 Outubro 2020].

[90] “ Programmable Search - Edit search engines,” [Online]. Available: https://programmablesearchengine.google.com/cse/all. [Acedido em 28 Fevereiro 2021].

[91] “Method: cse.list | Custom Search JSON API | Google Developers,” [Online]. Available: https://developers.google.com/custom-search/v1/reference/rest/v1/cse/list. [Acedido em 3 Março 2021].

[92] “Maltego,” [Online]. Available: https://www.maltego.com.

[93] Secret Intelligence A Reader.

[94] AllTech, “Run Python Script from C#,” 25 01 2019. [Online]. Available: https://www.youtube.com/watch?v=g1VWGdHRkHs. [Acedido em 27 Junho 2021].

[95] “Chapter 5. Automatic Graph Layout,” [Online]. Available: https://docs.yworks.com/yfiles/doc/developers-guide/tree_layouter.html. [Acedido em 27 Junho 2021].

[96] “Repositório Chameleon no Github,” [Online]. Available: https://github.com/ghostwords/chameleon. [Acedido em 27 Junho 2021].

.

Anexos

Anexo A

```
namespace FREE_OSINT_LinkedIn_Metadata_Extractor
{
    class Program : IGeneral_module, IProcessing_Module
    {
        private string title = "FREE-OSINT LinkedIn Metadata Extractor";
        private string description = "Extracts Name, Surname and Picture URL from profile metadata. Apply only on FREE-
OSINT_Google_Custom_Search LinkedIn results";
        private string extracted_node_title = "FREE-OSINT LinkedIn Metadata Extractor";
        static void Main(string[] args)
        {

        }

        public string Description()
        {
            return this.description;
        }

        public TreeNode Process(TreeNode treeNode)
        {
            TreeNode finalNode = new TreeNode(extracted_node_title);
            List<TreeNode> extracted_list = new List<TreeNode>();
            if (treeNode.Nodes.Count > 0)
            {
                foreach (TreeNode sub in treeNode.Nodes)
                {
                    TreeNode extracted;
                    extracted = extract_metadata(sub);
                    if (extracted != null)
                        extracted_list.Add(extracted);
                }
            }

            finalNode.Nodes.AddRange(extracted_list.ToArray());
            return finalNode;
        }

        private TreeNode extract_metadata(TreeNode subnode)
        {
            if (subnode.Nodes.Count > 0)
            {
                TreeNode link = null;
                foreach (TreeNode subsubNode in subnode.Nodes)
                {
                    if (subsubNode.Text.Contains("linkedin.com"))
                    {
                        link = subsubNode;
                    }
                    if (subsubNode.Text.Equals("Metadata"))
                    {
                        String first_name = "";
                        String last_name = "";
                        String image_link = "";
                        foreach (TreeNode meta_node in subsubNode.Nodes)
                        {
                            if (meta_node.Text.Contains("og:image"))
                            {
```

```
                    image_link = meta_node.Text.Split(new string[] { ": \"" }, StringSplitOptions.None)[1];
                            image_link = image_link.Remove(image_link.Length - 1);
                        }
                        else if (meta_node.Text.Contains("profile:last_name"))
                        {
                            last_name = meta_node.Text.Split(new string[] { ": \"" }, StringSplitOptions.None)[1];
                            last_name = last_name.Remove(last_name.Length - 1);
                        }
                        else if (meta_node.Text.Contains("profile:first_name"))
                        {
                            first_name = meta_node.Text.Split(new string[] { ": \"" }, StringSplitOptions.None)[1];
                            first_name = first_name.Remove(first_name.Length - 1);
                        }
                    }
                    if(first_name.Length > 1 || last_name.Length > 1)
                    {

                        TreeNode extracted = new TreeNode(first_name + " " + last_name);
                        extracted.Nodes.Add(new TreeNode(first_name));
                        extracted.Nodes.Add(new TreeNode(last_name));
                        extracted.Nodes.Add(new TreeNode(link.Text));
                        extracted.Nodes.Add(new TreeNode(image_link));
                        return extracted;
                    }
                }
                else if (subsubNode.Nodes.Count > 0)
                {
                    return extract_metadata(subsubNode);
                }
            }
            return null;
        }
        else
        {
            return null;
        }
    }

    public string Title()
    {
        return this.title;
    }
  }
}
```

Anexo 1 - Código do módulo LinkedIn Metadata Extractor