UNIVERSIDADE BEIRA INTERIOR
Engenharia

Dynamic OSINT System
Sourcing from Social Networks

Versão final após defesa

Mónica Amoroso Rodrigues

Dissertação para obtenção do Grau de Mestre em
Engenharia Informática
(2º ciclo de estudos)

Orientador: Prof. Doutor Pedro Ricardo Morais Inácio
Co-orientador: Prof. Doutor João Paulo da Costa Cordeiro

Covilhã, Janeiro de 2022

Dedicatória

Todo o esforço seria em vão se não juntasse razão e coração.
Luzinha, por vires mudar tudo para melhor.
Micael, pelos momentos de concentração e pelo apoio.
Pai, por investires em mim quando mais precisei.
Mãe, por nunca me deixares esquecer que havia um projeto pendente.
Jorge e Gonçalo, por aguentarem o campo de guerra em que se transformou a nossa casa.
Aos orientadores, pela oportunidade e paciência.
Aos colegas de trabalho, pela tolerância e compreensão.
A ti avó, por mandares sempre bolachas e pacotes de açúcar, precisei de bastante energia.
E por último e não menos importante, a ti avô, que a tua veia da determinação nos acompanhe sempre.

Resumo

No passado, era humanamente impossível observar e extrair grandes quantidades de informações textuais de plataformas da *web* em curtos espaços de tempo, mas a tendência mudou e nos últimos anos surgiram diversos sistemas de vigilância baseados na seleção e extração de informação textual proveniente de fontes de informação abertas, denominadas *Open-Source Intelligence* (OSINT), que se têm tornado populares principalmente entre os profissionais de segurança informática, permitindo a deteção de novas ameaças, a localização e recolha de informação disponível para o público em geral através das redes sociais, *blogs*, jornais, televisão, etc., revelando-se uma grande vantagem em termos de recolha de informação e uma boa ajuda no que diz respeito à prevenção de problemas principalmente na área de segurança da informação.

Esta dissertação foca-se no desenvolvimento de uma plataforma com base em informação *open source*, dando continuidade a um trabalho anteriormente desenvolvido numa outra tecnologia - Hypertext Preprocessor (PHP), onde se apresentaram fórmulas e algoritmos para classificação de *posts* do *Twitter* sobre o tema da segurança da informação. Focando-se este trabalho no desenvolvimento de novas versões da plataforma com base na tecnologia *Node JS*, na implementação das fórmulas apresentadas, na melhoria da experiência do utilizador (UX) e na avaliação da plataforma desenvolvida com utilizadores.

Durante o desenvolvimento do trabalho foram apresentadas duas versões da plataforma, e hospedadas numa máquina virtual, tornando-as acessíveis aos utilizadores, que na fase final contribuíram com o seu *feedback* sobre as mesmas. Essa máquina virtual baseia-se em serviços *cloud* da *Microsoft Azure*, onde estão instalados três processos desenvolvidos em *Node JS* (um que disponibiliza a página, um que classifica, e outro que recolhe *posts*), os *posts* são recolhidos através de uma API disponibilizada pelo *Twitter*, e guardados numa base de dados *MySql*, baseada na plataforma de administração de base de dados *PHP-MyAdmin*, disponibilizando à comunidade as notícias mais recentes e relevantes sobre vários temas.

Durante o processo de desenvolvimento teve-se em conta o modelo *User-Centered Design* (UCD), um processo focado no utilizador e na experiência de utilização. A participação de utilizadores foi assim a chave para a definição das características, e da forma como é apresentado o *front-end* da plataforma, sendo estes incluídos na fase de testes, com o preenchimento de formulários visando recolher *feedback* sobre os protótipos desenvolvidos.

Com base no *feedback* recolhido foram implementadas novas melhorias. De todas as mais relevantes foram: a possibilidade de pesquisa por vários temas em simultâneo, a inserção de monitores, e a possibilidade de aplicar filtros, como o número de minutos em que os *posts* ficam disponíveis no ecrã, e a ordem com que os mesmos devem ser apresentados.

Resumo alargado

Este capítulo representa um resumo alargado do presente documento, dando uma breve introdução ao tema, e apresentando o enquadramento, a descrição do problema e os objetivos. Descreve a plataforma sucintamente, e apresenta os resultados dos testes realizados. As principais conclusões e tarefas identificadas como trabalho futuro.

Introdução

As plataformas digitais tornaram-se populares, e são muito utilizadas para verificar notícias e informações sobre diversos assuntos de interesse dos utilizadores. Mesmo que se possam configurar assuntos/tópicos que desejam seguir, milhares de *posts* são inseridos diariamente, e não é fácil focar a atenção no que realmente importa, no meio de informações redundantes e notícias pouco credíveis. O Twitter como rede social permite que os utilizadores se atualizem sobre temas da atualidade em diversas áreas. Existem na atualidade alguns sistemas que permitem a pesquisa de informação sobre utilizadores, ou empresas, ou informações específicas, no âmbito das plataformas/sistemas baseados em Open-Source Intelligence (OSINT), mas de todos os que foram observados durante a fase de pesquisa e estado da arte, não se encontrou nenhum que fosse ao encontro de todos os objetivos da plataforma proposta.

Enquadramento, Descrição do Problema e Objetivos

O *Twitter* é uma plataforma útil para diferentes pessoas, com diferentes interesses de informação, visto que os *tweets* podem fornecer informações sobre vários assuntos, onde, por exemplo, notícias e questões problemáticas podem ser encontradas. No contexto da cibersegurança podem ser identificados os ataques mais recentes, tornando esta uma fonte de informação importante para pessoas que desenvolvem sistemas que podem estar em risco a determinadas vulnerabilidades, para serem tomadas medidas preventivas, com base em *OSINT*.

A primeira etapa para obter informações baseia-se em *crawlers*, conforme descrito em 2.4 e nas Interfaces de Programação de Aplicações (API). Após a recolha de dados é necessário filtrar e classificar a informação para obter apenas o que o utilizador necessita, recorrendo à Linguagem de Programação Natural (NPL).

Esta dissertação dá continuidade ao trabalho anteriormente iniciado, onde foram apresentadas fórmulas para classificação de textos e uma plataforma que representa os *posts* pela relevância atribuída por essas fórmulas. Contribuindo este trabalho com uma nova plataforma, que teve uma versão intermédia avaliada com utilizadores, e só depois foi desenvolvida e melhorada a segunda versão, que se denominou como: versão final. O desenvolvimento de ambas as versões seguiu um processo de *User-Centered Design* (UCD),

incluindo utilizadores nas fases de testes.

Em suma a ordem das tarefas realizadas no decorrer deste trabalho, tiveram como base a ordem dos objetivos apresentados:

- Estudo do impacto das alterações associadas às métricas utilizadas nas fórmulas;
- Implementação das fórmulas para a classificação de *tweets*, integrando-as e classificando-as em relação à versão anterior da plataforma;
- Melhoria da experiência de utilização da plataforma;
- Avaliação dos protótipos.

Principais Contribuições

Para dar resposta aos objetivos propostos foi desenvolvida uma nova plataforma, baseada em três processos (um que disponibiliza a página, um que classifica, e outro que recolhe *posts*), os *posts* são recolhidos através de uma API disponibilizada pelo *Twitter*, e guardados numa base de dados *MySql*, baseada na plataforma de administração de base de dados *PHPMyAdmin*, disponibilizando à comunidade as notícias mais recentes e relevantes sobre vários temas.

Os *posts* são classificados por fórmulas que se baseiam no texto dos *tweets*, sendo as palavras o foco desta parte da dissertação, visto que as mesmas foram guardadas numa tabela onde se regista o número de ocorrências e o número de *posts* em que aparecem, sendo que com base nestes parâmetros é feito o cálculo da probabilidade num determinado domínio. Quantas mais palavras forem inseridas melhor irá ser a classificação das mais relevantes e menor a das menos relevantes. As palavras relacionam-se com temas, dando assim a possibilidade de a plataforma se focar em vários temas configuráveis pelos utilizadores.

Plataforma Desenvolvida e Testes

Foram desenvolvidas duas versões da plataforma apresentada, a primeira era uma simples página de pesquisa, e que com base no *feedback* dos utilizadores evoluiu para uma versão mais completa, permitindo configurações como: os temas que os utilizadores podem seguir, o número de minutos em que os *posts* vão aparecer desde a sua recolha, e a ordenação dos *posts* apresentados na página por data ou relevância. A segunda versão da plataforma foi avaliada por utilizadores, alguns deles utilizadores do Twitter, recolhendo-se *feedback*, que da sua análise se obteve novas características e alguns pontos sobre trabalho futuro.

Tanto no final dos desenvolvimentos da primeira como da segunda versão da plataforma foram feitos testes com utilizadores. Sendo enviados formulários que continham perguntas sobre experiência de utilização e onde foi solicitada a opinião dos utilizadores. Foram também feitas entrevistas com o mesmo objetivo, e no final aplicou-se a técnica de *card sort* para agrupar a informação recolhida com o intuito de remover características ou problemas apontados parecidos, ou idênticos, ou que fossem ultrapassados com outras soluções.

Trabalho Futuro

Alguns recursos que podem representar melhorias futuras foram identificados pelos próprios utilizadores durante a avaliação da plataforma. Recolhendo-se os comentários efetuados, os seguintes pontos foram marcados como importantes num futuro próximo:

- Link para o *post* do *Twitter*, permitindo a navegação entre a plataforma desenvolvida e o *Twitter*;
- Seleção do idioma, permitindo que o utilizador tenha à sua disposição uma nova configuração onde poderá mudar o idioma no qual pretende que os *tweets* sejam mostrados;
- Mostrar as imagens e vídeos associados ao tweet;
- Ao clicar num monitor permitir que surjam *posts* do espaço de tempo a que esse monitor se refere;
- Melhoria da *Escalabilidade* e *performance*, para que a plataforma possa chegar a muitos utilizadores;

Conclusões

A informação é necessária para a tomada de decisões em muitas áreas. Inicialmente esta dissertação focou-se apenas nas áreas de segurança informática, sendo o protótipo apresentado uma ferramenta útil para recolher e filtrar informação útil para os engenheiros de segurança, ajudando-os na tomada de decisões sobre novos ataques e novas técnicas para prevenir problemas de segurança. Mas muitas outras áreas podem ser beneficiadas com o uso de um sistema como este, como jornalismo, educação, medicina e defesa.

Durante o desenvolvimento das duas versões da plataforma, a opinião dos utilizadores foi muito importante para saber o que seria importante incluir, o que é fácil de usar e o que não traz nenhum valor para o utilizador. Os utilizadores que testaram a plataforma foram principalmente informáticos, mas foram consultadas também outras pessoas, das áreas de artes, história, gestão de empresas e finanças, visto que a plataforma é dirigida a vários públicos.

A primeira versão desenvolvida, como uma versão simples do que foi proposto e apresentado, foi importante para testar e melhorar as fórmulas e verificar a classificação aplicada, então com base na opinião dos utilizadores foi possível melhorar a plataforma tendo a versão final apresentada, onde mais recursos foram encontrados para serem implementados e marcados como trabalhos futuros.

Abstract

In the past, it was humanly impossible to observe and extract large amounts of textual information from *web* platforms in short periods of time, but the trend has changed and in recent years several surveillance, selection, and extraction of textual information systems have emerged, based on *Open-Source Intelligence* (OSINT). These platforms became popular among computer security professionals, allowing them to detect new threats and respond in a timely manner by locating, collecting and analysing information made available to the public through social networks, *blogs*, newspapers, television, etc., proving to be a great advantage in terms of information gathering and a good help with regards to preventing problems, especially in the area of information security.

This dissertation focuses on the development of a platform based on *OSINT*, and has two main objectives. First, to continue the work previously developed in another technology - Hypertext Preprocessor (PHP), in which formulas and algorithms were developed to classify *posts* from *Twitter*. And second, to present a new platform (using Node JS technology), by applying the formulas from the previous work, evaluating the new platform with users, and improving the user experience (UX).

During the development process two versions were provided to the users and hosted on a virtual machine, based on *cloud* services of *Microsoft Azure*. The platform architecture is composed by three processes developed in *Node JS* (one that provides the page, the web server, one that collects the *posts*, and another one that does the classification of each *post*). The *posts* are collected through an API provided by *Twitter*, and stored and managed in *PHPMyAdmin* a platform based on *MySql* database.

The *User-Centered Design* (UCD) was applied during the development process, a process that is focused on the user and his experience. The participation of users has contributed to define new features and to improve the presented layout. Users were included in the testing phase, being called to fill forms, one form for each version.

Based on the collected *feedback*, the following improvements were implemented: the possibility of searching for several topics at the same time, the possibility of havving header monitors by ranges of time, and the possibility of applying filters, such as the number of minutes the *posts* are available on the screen, and the order by which they are presented.

Keywords

Crawlers, Information Security, Open Source Intelligence, Peer-to-Peer, Tweets, Twitter, Usability Testing, User Centered Design, User Experience, Web Development

Contents

List of Figures

List of Tables

List of Listings

Acronyms List

API	Application Programming Interface
CSS	Cascading Style Sheet
DBMS	Database Management System
DNS	Domain Name System
IA	Information Architecture
IP	Internet Protocol
JSON	JavaScript Object Notation
HTML	HyperText Markup Language
Node JS	Open-source platform based on JavaScript runtime environment
OSINT	Open-Source Intelligence
P2P	PeertoPeer
UBI	Universidade da Beira Interior
UCD	User Centered Design
UI	User Interface
URL	Uniform Resource Locator
UX	User Experience
WWW	World Wide Web

Chapter 1

Introduction

1.1 Scope and Motivation

Digital platforms are popular nowadays, and very used to check news and information about various topics of the user interest. Even if user can configure the information subjects that he or she wants to follow, thousands of posts are inserted every day, and it is not easy to focus attention to what really matters, in the middle of repeated information and fake news.

This dissertation is focused on *Twitter* users, and the information that they can absorb from this platform. *Twitter* is a useful platform for different people, with different information interests, tweets can provide information about various topics and identify problematic issues. For example, in the cyber-security context the most recent attacks can be identified, and the responsible persons for security in the companies being informed about these topics and take preventive measures, based on the information provided by these systems.

Related to information gathering, crawlers as described in 2.4 are a relevant technology in data gathering, as Application Programming Interfaces (APIs), that allow to get information from a given source. After the information is gathered, the information needs to be classified, in this context refers to tweets text, that should be analyzed, based on Natural Programming Language (NPL), in order to classify the text posted by the user.

This dissertation is the continuation of the work started by Ivan [Fer16], that presented a platform based on formulas to classify posts from Twitter. This work started by the revision of the formulas presented by him, and the implementation of a new platform based on Node JS technology. The first version of this platform contained the same features as the ones presented by Ivan. This first version was evaluated by user, and based on the feedback provided, new features were implemented during the development of the second version. All the development process was based on user by implementing User-Centered Design (UCD). Users were asked to provide their opinion by filling a form in the end of each version. That means that two forms were sent to users in order to gather *feedback*.

In this work, the only source of information was defined to be *Twitter*, with other information sources not being considered for the objectives proposed for this master dissertation, but the architecture of the platform is prepared to receive new sources of posts, by implementing new gathering processes in the virtual machine, that do not need to be im-

plemented in Node JS, only need to have access to the database and insert the posts in the actual database architecture format. After the new posts from other sources are inserted, the evaluation will be done the same way that is done for Twitter posts, by attributing a relevance to each one of them.

1.2 Problem Statement and Objectives

As this work is the continuation of the provided platform by Ivan [Fer16], and this platform was not evaluated, the main objective is to focus on user and evaluation of the platform, with the main objectives being:

- Study and review the formulas;
- Assess if new metrics are useful for the classification of *twitters* and posts from users, integrate them accordingly, and study the impact of their inclusion against the previous version of the system;
- Improve the overall look and feel of the platform;
- Evaluate the prototyped system with end users.

1.3 Document Organization

This document is structured in five chapters. The chapters are as follows:

1. The **first** - Introduction - chapter presents the scope and motivation, problem statement, objectives and how the document is organized;
2. The **second** - Open-Source Intelligence Systems and Twitter - chapter, describes the framing of the Social Networks, the importance of technological footprint, defines and presents some open source information (OSINT) Systems, the Datafication and NLP and describes the Crawlers types and when they are mostly applied;
3. The **third** - User Experience - chapter is where the user experience techniques are detailed, focusing on techniques that were important for the development process;
4. The **fourth** chapter - Metrics and System Proposal - is where the implemented formulas are described, and the developed platform, the used technologies and tools, and the tests and forms that were made by user are presented;
5. And the last, the **fifth** chapter, presents the conclusions and the future work.

At the end of the dissertation, five appendixes, containing important information to complement the discussion, are included:

- Visual Design Principles;
- Accessibility Guidelines;
- *Twitter JSON* Structure;
- First Version Evaluation Form;
- Second Version Evaluation Form.

Chapter 2

Open-source Intelligence Systems and Twitter

This dissertation is focused on OSINT, Social Networks, and how the information that may be useful for a wide range of purposes is provided. In this case, the technological *footprint* is described as an accumulation of information that can bring good results and be very advantageous, but also the basis of personal security issues, and pose at high risks those who share personal information on Social Networks.
Otherwise, other processes, as well as marketing initiatives, or police investigations rely on people information, from their opinions and interests, free information can become very important and be the basis of studies and reports for many purposes.

From this collection of intelligence [(Ed07] arises the term *OSINT* and of course the *datafication* process that allows transforming the collected information in readable data, based on natural language processing, a term that is related to Social Network posts analysis, since the way that people communicate depends on the culture, the country, the language that the user is using in their posts, the age, and other factors.

The final part of this chapter focuses on crawlers, where the types of crawlers and several algorithms are described.

2.1 Social Networks and Technological Footprint

Technological *footprint* is the term that defines the information left by people, when they use Social Networks and leave personal information publicly available.

In this section the advantages and disadvantages of leaving information publicly available are described, what can be done for personal data protection, and the laws that protect European and Portuguese users.

2.1.1 Capabilities and Limitations

A consequence of technological *footprint* is that society constantly uses social networks, leaving personal information, opinions and sharing experiences with friends and with the public in general. This information is considered an important input for studies in varied areas, but mainly for marketing effects, for example, to suggest products or advertising. Companies can more easily offer incentives based on interests and needs, resulting in cost savings. On the other hand, getting updated news is nowadays much more than reading a newspaper. By using social networks it is possible to find journalists or considered informed people sharing relevant news. Following these people is a good way to keep in-

formed.

In the context of fraud or legal issues, irregularities can more easily be detected, as the user *footprint* can be followed using information left in the Internet. The main disadvantage of leaving personal information available, is that this data can be used without the consent of the respective users, causing annoyance, for example in marketing campaigns. Still in the field of marketing, data collected by a company may be prejudicial for users in different ways, for example, buying habits or account balances may lead banks to define users profile and assign higher credit rates.

2.1.2 Personal Data Protection

It is known that social media platforms like Facebook are used daily by millions of people [Nic17c] for different purposes, such as socialization and information, and also for business, from different devices. The most usual used device is the smartphone, since according to [Nic17c] 94 percent of Social Network users use smartphones, next, the most used is the laptop and the less used is the desktop. There are many reasons to use Social Networks, and the risks implicit in its use are more acceptable for some users than others. Still, according to [Nic17c], more than 41 percent of the respondents of the questionnaire are not conscious about exposing their personal information in social networks can be dangerous.

Some procedures to avoid problems while using social networks are:

- Change password with regularity;
- Do not share personal information with unknown people;
- Report any abuse.

2.1.3 Personal Data Law

There are some entities in Portugal and in Europe that can help in cases of abuse, although in some cases the law is not defined. In Portugal, General Regulation of Data Protection, in Portuguese: *Regulamento Geral de Proteção de Dados* (RGPD), was applied on May 25, 2018, and refers to the following subjects:

1. Information to data subjects;
2. Exercise of the rights of data subjects;
3. Consent of data subjects;
4. Sensitive Data;
5. Documentation and Registration of treatment activities;

6. Subcontracting contracts;
7. Data protection Responsible Person;
8. Technical and organizational measures and treatment safety;
9. Protection of data from design and impact assessment;
10. Notification of security breaches.

2.1.4 Sharing Online Content

Still in the data protection context, a wide range of special categories of data exists, including biometric data, which became part of the list of sensitive data.

Because of sharing sensitive information, every user should answer the following questions before making some information available:

- Who can access the information I am disseminating online?
- Who controls and owns the information I insert into a social networking site?
- What information about me are my contacts passing on to other people?
- Will my contacts mind if I share information about them with other people?
- Do I trust everyone with whom I am connected?

2.2 OSINT

Information circulates freely on the Internet, and can be found for all the subjects. *OSINT* term arises in open-source information context, in this case information from posts shared by people on Social Networks, from where intelligence can be obtained and analyzed for many purposes. OSINT crosses all boundaries, and can be used with the purpose to save lives, time, and costs.

2.2.1 Getting Only The Important

In the context of OSINT, the abstraction concept is essential, in order to retain only the important from all the information that can be searched, this technique is designated by Situational Awareness [CO17], and involves to be aware of what is happening around, in order to understand how information, events, or in a most particular sense, our own actions will impact goals and objectives at medium and long-term. The worst errors, or in the case of information security, the worst attacks have more chances to occur if they are not identified, or if no one is alert for them to happen. This can be applied to any subject, even during quotidian life, e.g., to ride a bicycle we need to be alert to the dangers that could occur while always being focused on the most important, the fact that we are riding a bicycle. In the OSINT context the same occurs, the information needed for some purpose should be entropy aware.

2.2.2 Capabilities and Limitations

In order to get only relevant information, the main problem lies in the selection, which means that not everything found is reliable, or from a safe source, being this the main problem of this subject. Researchers and journalists working in the field, for example, are a valuable source of human intelligence. An example of this is the information shared by them, and from other people too, but mainly by these people as they are sources of reliable information.

The main characteristic of open source is the high availability and constant up-to-date sources. One example of these sources is Twitter. Information gathered from open sources is a great resource for intelligence and can be used to support the creation of strategies or reports for a variety of purposes.

2.2.3 Historical Facts

> ”Intelligence means every sort of information about the enemy and his country.”
>
> *Clausewitz, On War, 1832*

Contrary to the real meaning of the word OSINT, which treats open-source information, its origins are entirely linked to the secret information, as referred by Florian Schaurer and Jan Störger [SS13].

In the last centuries, the terms of intelligence and secrecy were linked to the high society people, and to kings and states, and these were used as another type of war of spies and counter-spies, taking advantage of covert actions and plausible deniability. As referred by Florian Schaurer and Jan Störger [SS13] the movement that originated the OSINT started in the United States. Strategic analysis has been highlighted by Sherman Kent during the Second World War when he promoted the theory, doctrine, and practice of intelligence analysis.

The Office of Strategic Services (OSS), the Central Intelligence Group (CIG) and the Central Intelligence Agency (CIA) developed the *OSINT* concept as we know it today. Later the topic was addressed in the National Security Act, in 1947, by president Harry Truman. Focused on a major restructuring of the United States government military and intelligence agencies just after World War II, not being a very relevant term at the time and coming to have a greater focus later on. In 1988 a great evolution, initiated by the movement of Collective Intelligence (sometimes called “smart mobs”, “wisdom of the crowds”, “world brain”) was pronounced by several authors and scientists of the time, among them Qunicy Wright, as well as the creation of the Foreign Broadcast Monitoring Service (FBMS), an agency responsible for the monitoring of foreign broadcasts.

As referred by Florian Schaurer and Jan Störger [SS13] the Aspin–Brown Commission, known as the Commission on the Roles and Capabilities of the US Intelligence Community, has been important since they called attention to the “severely deficient” access to open sources in **1996**. In **2004**, due to the September 11 attacks, a Commission called 9/11 (eleven September) recommended the creation of an *OSINT* agency.In the beginning of **2005**, the Iraq Intelligence Commission recommended the creation of open-source guidelines at the CIA. After these recommendations the DNI Open-Source Center emerged, still in the same year, 2005. The main objectives were based on the collection of information available from the Internet, databases, press, radio, television, video, geospatial data, photos and commercial imagery.

In **2006**, Eliot A. Jardines as Assistant Deputy Director of National Intelligence for Open Source (ADDNI/OS) has established the National Open-Source Enterprise (OSE) and

authored intelligence community directive 301, that outlined responsibilities and established policies for the intelligence community regarding open-source intelligence activities.

2.2.4 OSINT Existing Frameworks

The use of some existing platforms can be helpful, but sometimes is not enough, since each of them are not designed to "serve" all the purposes, being incomplete, or completing each other, transforming the development of OSINT frameworks in an ongoing problem for many companies, and being considered an actual engineering task [Ber14], that is in constant evolution.

There are very powerful tools and frameworks with different characteristics, maybe because each one of them is designed for a single purpose, and not to be adapted to many requirements.

In a page called *OSINTFramework* [Jus17] is possible to find a tree with many subjects, that can provide some help in this sense since it gives a possible solution, that is, it redirects to pages with the solutions available for the type of data that needs to be obtained.

Due to space constraints, it is not possible to show or discuss with detail all the characteristics of each of the OSINT tools, but a small subset of them will be presented below. These tools follow in two main areas: the collection of information about **infrastructural reconnaissance** and **personal reconnaissance**s.

2.2.4.1 Maltego

Maltego [Pat17] is a framework that provides information about both infrastructural reconnaissance and personal reconnaissances.

It is a Java tool, available in desktop application. The main purpose is to serve companies, and it is available for four types of clients, has a free version, and paid versions, with prices varying according to the type of client selected.

The companies can use it on their infrastructure to gather sensitive data about the target organization, email addresses of employees, confidential files which are handled carelessly, internal phone numbers, DNS records, IP address information, geolocation of the network, in addition to many other features.
For the personal area, it enables to harvest of person-specific information, such as social networking activity, email addresses, websites associated with the person, telephone numbers, and other information that can be configured on this tool.
All the data that can be obtained by Maltego, in the most of the client types, can be converted to a spreadsheet, in a report format, to then be easily analyzed by the management

of the company.

2.2.4.2 Shodan

Sentient Hyper-Optimized Data Access Network (Shodan) [sho17] is about infrastructural reconnaissance and counts with three paid account types with different characteristics. Being a search engine to crawl information that is shown in a web page. Shodan has servers located around the world that crawl the Internet, constantly providing the latest Internet intelligence.

Information like location, server version, and software installed, after the search all information can be converted in a report, that can then be analyzed, e.g., by companies, to help identify security issues in their servers. Technically it integrates with other platforms, such as Maltego [Pat17], FOCA [Ele17], Nmap [nma17], and many others. In terms of development, it has libraries in Python, Ruby, PHP, C#, Go, Haskell, Java, Node.js, Perl, PowerShell, and Rust. The service provided uses representational state transfer (REST) API or a Streaming API.

2.2.4.3 Metagoofil and FOCA

Metagoofil [Chr17] is an information gathering tool based on Google search engine, used for extracting meta-data from public documents, PDF, DOC, XLS or PPT.

The procedure that occurs starts by a search for files in a target domain by using Google search engine, then all the documents found are saved to the local disk, and the meta-data is extracted, finally a result report is generated, and presented in a HTML page.

One of the most valuable features is that this tool searches in Microsoft Office files, from where it is possible to get the kind of network hardware that is used at the target installation, the type of operating system, or the network name, disclosed PATH, shared resources, even *usernames*.

To use it Kali Linux [KAL17] is needed, and after installation, all that is required is a command line.

In the case of Fingerprinting Organizations with Collected Archives **(FOCA)** tool [Ele17] is a metadata file analyzer, that is usable for:

- Metadata extraction;
- Network analysis;
- DNS Snooping;
- Search for common files;

- Juicy files;
- Proxies search;
- Technologies identification;
- Fingerprinting;
- Leaks;
- Backups search;
- Error forcing;
- Open directories search.

Both tools collect information about infrastructural reconnaissance.

2.2.4.4 GHDB

Exploit Database [Exp17], mostly known as Exploit DB, is an archive of public exploits and corresponding vulnerable software, developed for use by penetration testers and vulnerability researchers, but probably also for hackers and attackers. This is available for free.

Google Hacking Database (GHDB) [Goo17] is based on the original exploit database, and here is possible to search by vulnerabilities or the known security issues available in this database.

This search could be used to find *usernames*, passwords, e-mails and other information, and is done by these categories:

- Footholds;
- Files Containing Usernames;
- Sensitive Directories;
- Web Server Detection;
- Vulnerable Files;
- Vulnerable Servers;
- Error Messages;
- Files Containing Juicy Info;
- Files Containing Passwords;
- Sensitive Online Shopping Info;

- Network or Vulnerability Data;
- Pages Containing Login Portals;
- Various Online Devices;
- Advisories and Vulnerabilities.

This tool is free and does not require installation, the only requirement is to have a browser.

2.2.4.5 Social Engineer Toolkit

Social-Engineer Toolkit [Sec17] is an open-source tool, that is used to perform online social engineering attacks. The tool can be used for attacks such as: spear phishing and website attack vectors. It is possible to enable the execution of client-side attacks and the harvesting of credentials. For example, an executable with a fake login page can be sent to the victim and then the information provided by the victim is saved on a server; this information can be *username*, email, phone number, password, address, or others.

2.2.4.6 Conclusions

The importance of safeguarding information about people and companies is revealed by the use of these tools, which can be used to protect the data of a person or a company, verifying in which places certain information is being reached, and what type of information is being shared unduly, illustrating the need to review shared information and safeguard personal information.

2.3 Datafication and Natural Language Processing

"Knowledge is power."

Francis Bacon

Information comes from everywhere, from people data, to weather data, to historical events, and others that are converted in data every day, or maybe every second. This process is called *Datafication* [bMTS17] and arises from the transformation of information in useful data, in an effort to making it ready to get meaningful answers for the most varied questions.
Typically, the information used by OSINT frameworks is information that is online, and already passed through a "datafication process" [bMTS17].
Information is important for companies, to help to make decisions, being necessary an analysis from internal(related to the company business) and/or external data (from clients for example). Companies use information to make better decisions in order to define processes and strategies, depending on data intelligence to evolve.

2.3.1 Datafication Process

Datafication occurs after the Natural Language Processing, characterized by multiple techniques, as well as Tokenization and Sentence Segmentation, Parsing Techniques, Lexical Analysis and Semantic Analysis.
There is a datafication process that usually is followed:

1. This process starts by the definition of the **sources** or places where the information can be found, assuming that the information is available in distinct sources;
2. Then the **harvesting** occurs, where the relevant information from the identified sources is collected;
3. After data collection the information is **processed** in order to get only the meaningful information;
4. When there is more than one source, a **merge** of information usually happens, and the information is joined in an only place, where it can be later consulted;
5. The final step is to make information legible, normally this info comes in a **report**, or various reports since multiple questions can be done for the stored data. This step is the most important for the decision makers in a company.

2.3.2 Datafication Chalenges

While the process seems perfect, some problems may occur, compromising the final results, especially when the information is public. As a source of reliable information, the

Internet must be approached with great caution, as some challenges emerge, like Confidentiality, Integrity, Availability, Possession, Authenticity and Utility, that should be taken into account. These challenges are detailed below:

1. **Confidentiality** means that only authorized people should have access to certain information.

2. **Integrity** represents the consistency of information, that should be unchanged, and not corrupted, being this one of the biggest problems.

3. **Availability** ensures the timely and reliable access to information by holders of the appropriate security clearance.

4. **Possession** means the legal ownership and physical control of information should be accomplished.

5. **Authenticity** represents the veracity of the origin and authorship of information

6. **Utility** verifies whether the information is useful for the intended intelligence purpose.

2.3.3 Digitization

With the increase of the Internet usage, more and more multimedia communications and information sharing are taking place. Information on the Internet leads us to believe that this source has everything that is necessary, but sometimes this is not true. Much information that may be necessary has not yet passed by the **Digitization** [(Ed07] process. But even so not all are disadvantages, given that most updated information from the main and niche media as well as publishers and bloggers or individuals are currently available, as well as the historical information, including policies and financial statements of great importance to specific nations, industries, organizations and tribes, can now be found Online, and provide a good support for companies.

Besides this, the constant share of information in social networks, becomes a way of being updated, replacing the necessity of consulting newspapers, in this way the news are shared much quicker than by medias, this phenomenon of sharing is called **Peer-to-Peer (P2P)** [(Ed07].

Sharing information can be divided in two perspectives, the part of the information that we want to share, and the other information that we do not even know that we are sharing, but most of the time we have agreed to share it without having noticed. This can be a problem to the person that shares it, but for companies this information is very valuable.

2.3.4 Secret Data

In the context of information sharing arises the secrecy when some information, needs or must be a secret. Secret information is information that needs to be available for some people only. But there are some different types of secrets characterized by two categories: the self-regulated, and the externally regulated secrets. The first ones are those that are not controlled by anyone and that belong to its keeper, and the seconds, the externally controlled are characterized by having someone who controls the secret.
In the self-regulated secrets:

- **Embarrassment secrets**: The ones that are motivated by embarrassment, or motivated by the fear of what others can think or do when the secret is revealed. Normally that leads the keeper to cause himself additional harm in attempting to avoid the revelation of the secret.
- **Control Secrets**: is the kind of secret that is usually kept achieving ends, where s fits the famous Portuguese phrase: "The secret is the soul of the business".
- **Privacy secrets**: kind of secret that characterizes by not giving the opportunity or the right of a given knowledge to others, the same as omission.

And in the externally controlled arise the:

- **Legal Secrets**: law or policy secrets, these should be inconsistent and not very well regulated, leading the keeper to doubts.
- **Social cohesion secrets**: there are secrets whose purpose may be of little value, to all who hold the secret, and to any stranger. When is lost, it can be easily replaced, the value is not in the secret itself.
- **Tradition secrets**: Without having a reason, a basic feature and a secret need to continue to be kept. Bureaucracies are especially suited to this kind of secrecy due to the lack of a mechanism for periodic review and review of regulatory and cultural structures.

2.4 Crawlers

The vastness of cyberspace requires automated tools to filter important content for one or more given purposes, that is what this section will describe, the Web Crawlers (or spiders), namely software programs that track information available on the Internet and stores it in local collections of data, or in spite of storing data, create indexation to large amounts of data that will later be referred to by a topic-driven crawler. A web crawler should make autonomous decisions about the progression of navigation.

In many cases the collected data is critical content for crime prevention or counter-terrorism, among other sensitive issues, this way data crawlers should be very efficient, not only on speed but also by the quality and credibility of the obtained data.

2.4.1 About Crawlers

As referred before, a Web Crawler is a program that tracks information on the Web and stores it in a local database or creates indexes in order to facilitate the work of a topic-driven crawler[RZ14].
The information on the network is disseminated through millions of pages, hosted on millions of servers. The user may access the information through existing hyperlinks, allowing it to move between existing pages. Similarly, a crawler can visit many pages in order to collect information that can be later analyzed and explored.
The dynamic nature of the web implies that it is not enough to get the information and keep it in a repository. Crawlers need to be kept informed about the pages and links which have been amended. This type of program has a wide range of practical applications:

- Business Intelligence: Obtain information from competitors and collaborators;
- Monitoring of interesting web pages: the user is notified when there is new information;
- Search Engine Support: Collects the information that will be indexed by the engine.

The essence of a crawler is in the way it gets information, the following are the main functions adopted by a crawler:

- The list is initialized with the seed URLs;
- In each iteration, the crawler selects the next URL on the border (list of URLs that still were not visited by the crawler);
- Search the corresponding URL page through HTTP;
- Scans the page and extracts URLs;
- Adds the new URLs to the crawl frontier and saves the page.

In addition to these features, that shows the basic operation of a crawler, others can be included to enable the crawler to perform more complex tasks such as page validation, structural analysis, and visualization, update notification, web mirroring and personal assistants/agents [Fer16].

The crawling process is not only tied to web content, it can act based on other sources. In this way, it cannot be restricted by a specific protocol, such as HTTP. It can work on SMTP, FTP, among others. Through the chosen protocol, the agent then makes a request to the data source, which ends up returning the desired content, from which the information will be extracted. This is to say that it is also possible to get information from documents, based on regular expressions, for example.

2.4.2 Web Mining

The concept of the Web crawler is directly tied to the concept of Web Mining, given that a Crawler handles data, as this section will describe the Web Mining theme. Web Mining involves techniques for recovering information, statistics, artificial intelligence and mining of data. It is a way used to discover and automatically extract data from services and documents available on the Internet, in general, the main web mining tasks are:

- **Document Search:** Information Retrieval that consists in the process of extracting data from of websites using the content of documents HTML, navigating between keywords and tags by eliminating them and retrieving the texts contained in these structures.

- **Selection and pre-processing:** pre-processing involves any kind of transformation of the information obtained in the search, for example, cutting texts.

- **Generalisation:** making use of artificial intelligence and data mining, the task of automatically discovering or one of several websites.

- **Analysis:** the analysis is generally applied in all the tasks of validating and interpreting the mined patterns.

Web mining can also be seen as part of the information retrieval process, as it can help to index, search, and rank documents. For example, clustering (a technique often used in Data Mining) can be used to index similar documents.
Web mining can be divided into three sub-areas:

- **Web Content Mining:** Is different from text mining because of the semi-structural nature of the Web, while text mining focuses on unstructured text. Web content mining, therefore, requires creative applications of data mining and/or text mining techniques as well as their own unique approaches.

  Textual databases usually contain long phrases or paragraphs, such as alert messages, reports, notes, or other documents. At textual databases may be unstructured (e.g., web pages), semi-structured (e.g., e-mails) or structured (e.g., library) [JH12].

- **Web Structure Mining**: is based on spiders by scanning websites, retrieving a homepage, and then linking information by reference to produce a specific page containing the desired information.

- **Web Usage Mining**: uses automated apparatuses to reveal and extricate data from servers and web reports, and it permits organizations to get both organized and unstructured information from browser activities, server logs, website and link structure, page content and different sources.

2.4.3 Crawler Types

Based on the analysis performed by Bing Liu [Liu11], crawlers can be defined in three main types: universal crawlers, focused crawlers, and topic crawlers. They are presented followed by each of these types.

- **Universal Crawler**

  To better understand the concept of the universal crawler, it is necessary to understand what a crawler is, and that it uses the first-in-width strategy. Thus, a *crawler breadth-first search* is one that has a border implemented in a First-In-First-Out queue (FIFO), indicating that a new URL is added to the queue tail and the next one to be crawled is the one which is at the head of the queue. Thus, a universal crawler differs from the first crawler in terms of performance, since it needs to search and analyze thousands of pages per second. With regard to the policy used by the universals, these try to encompass the most important web pages while keeping their indexes as up-to-date as possible.

- **Focused Crawler** [RZ14]

  Focused crawlers are used when searching pages in a given category, trying to skew the search according to the user interests, not being made a search for all web pages. In [SC99] a crawler of this type is proposed. In this solution, two key elements are used to guide the process: a classifier, which evaluates the relevance of a web document to the topic in focus, and a distiller, whose mission is to identify pages considered as good access points for many other relevant, designated hubs. It should be noted that the topics to be investigated are to the system in the form of examples that will be apprehended by the classifier. To perform the classification a base taxonomy is used. This proposed model is capable of collecting relevant information and identify popular resources, as well as regions with high relevance. It was found to be robust in different initial situations and demonstrated to be able to find good data away from the site of origin. Comparing this method with a normal crawler, that normal ones tend to get lost with existing noise, even starting at the same points of departure.

- **Topic Crawler**

  Topic crawlers do not have text sorters to guide them. The topic can be one or more example pages or even a small query. They are useful when examples of classified pages are not available in sufficient numbers to train the focused crawler before beginning the process.

Besides of the type of Web Crawlers, its behavior results from a combination of several policies [Cot04] such as:

- Selection policy, which states which web pages should be visited;
- Re-visit policy, which determines when the crawler should check for changes on pages;

- Politeness policy, which defines how to avoid overloading the servers responsible for the pages;
- Parallelization policy, that defines how to coordinate the operation of distributed Web crawlers.

2.4.4 Focused Web Crawlers

Focused Web Crawlers will be the base of this dissertation work. This subsection will detail this type of Crawlers.
As referred before, automated tools must be able both to inspect the actual contents of web pages and to make autonomous decisions about the navigation progression. According to Viv Cothey [RZ14] architecture of a Focused Crawler is composed by:

1. **Input:** a set of starting URLs and the target description are introduced into the framework;
2. **Web page downloading:** for each crawled page, the links are extracted and placed into a priority data structure;
3. **Web page content analysis:** web pages are analyzed to assess the content relevance with respect to the targets;
4. **URLs priority queue reordering:** the list of Web links to follow is rearranged, according to the computed relevance;
5. **Recursion:** URLs for link expansion are selected and steps 1) – 4) are repeated until stop criteria are fulfilled.

The described architecture will be clarified in the next topics:

Starting by the **URL Selection Strategy**, the core engine resembles a conventionally focused crawler represented on the figure 2.1, in the dark gray area the profile parameters and in the light gray the crawler engine [RZ14] are visible, in which a “scrapbook” holds the sorted list of URLs that have been encountered during web surfing, and that have been selected for future examination.
The following are methods of optimization for the URL selection task:

- Probabilistic methods;
- Link analysis;
- Structural approaches.

Each link retrieved in a downloaded page is assigned a score, measuring the relevance of the web-page that originated the link itself. The **content analysis process** relies on the general concept of “document”, which may include text files, web pages, and multimedia contents such as images. But in this case, the most important is the text analysis.

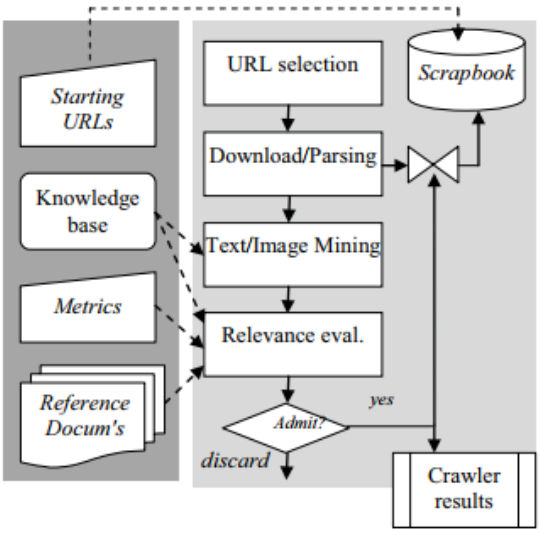


Figure 2.1: Schema of a Focused Crawler, presented by Viv Cothey 2.1.

Emil Gatial et al. [EGH99] presented the calculation of the **relevance of a web page**, which can be calculated according to the following formula:

$$R(page) = Sum(W(kw))/Nkw(page), \quad (2.1)$$

where R (page) is the relevancy rate of the page for a given domain characterized by a set of keywords, W (kw) is the weight of keyword "kw" and the Nkw(page) is the number of keywords found in the content of the page. The sum is evaluated for every keyword "kw" found in the content of the page. The value of predefined thresholds, whether the page is relevant or not. However, the value of W (kw) depends on a number of pages evaluated the relevant and irrelevant, the convenient value of threshold and the set of keywords could separate the miscellaneous pages into groups of relevant and irrelevant for given domain.

The final **ranking and admission** task involves the matching between the semantic description of the result and that of the reference documents; if conceptual domains do not overlap, the URL is discarded. This step is used to remove ambiguities in lexical contents. Browsing history is saved for admitted URLs, thus preserving the possibility to reconstruct the navigation path leading to a specific result. Whenever a web page is ranked as relevant, all hyperlinks contained in that document are scored accordingly and inserted in the scrapbook list for future download. Link-analysis and prediction mechanisms can apply at this step to pre-select the links that originate from relevant web pages.

2.5 Conclusions

The activity of filling out a simple form can mean the sharing of sensitive information with people that we do not want to share, making user sensitization important. In developers context the platforms and applications should be secure for users, as to prevent the main attacks in order to protect users data.

In other context, user data are useful for many areas as Marketing and the prevention of major incidents, such as terrorism, in the sense that shared information is analyzed in order to find something wrong.

There are many more tools that allow to search for data from people, systems, infrastructures, and businesses than the ones previously presented, this is a very current and ever-growing area and the number of tools keeps growing, since new tools are emerging every day. From all the presented tools the most complete one is the first, Maltego, Maltego, but it all depends on from many factors, and for the purpose the system is designed. *OSINT* is a wide area to discover, from the number of platforms to the definition of the subject. It is a thematic in constant grow, since even more information is available every day, and fake news and redundant information grows exponentially.

Converting data from the natural world to the digital is a necessary process. It is sensitive to errors, once an error can insert wrong or unreadable information, making the final product unclear. Users may be in this case susceptible to receiving incorrect information, or not receiving any information at all. These are systems that require a lot of attention to detail, and require a lot of testing, before being implemented in a production environment.

Automated search brings benefits over data search. Ensuring that the data will reach a certain destination to be used, and analyzed later. Being seen as an accelerator of data collection, their performance depends on the amount of data and the sites they are learned to track. In this sense, it is necessary to be careful with the sites where information is sought, as crawlers may consume private information and thus can incur problems for the person or company doing the research.

Chapter 3

User Experience

> ”Management is our universe, the consumer its center, and imagination its boundary”
> *Gerald Zaltman*

3.1 User Experience Overview

UX initials extend from **U**ser e**X**perience [NN20], this subject focus on how the end-users feel about using a given product (Webpage, Mobile App or a material product for example). When a new project starts, and when UX process is applied, normally UCD [Low13] is applied. This is not a rigid process, team by team, and project by project the process could change, and be more focused on some parts of the process than others, depending on the understanding of the UX team, and the objectives of the product. User experience means that the user is happy with the final product and that this one respects and exceed the expectations, in the way that the user feels that the product is perfect for him. This subject area impacts multiple disciplines, where technology, psychology, design, and engineering are included and are applied to product projects of many distinct types, such as objects, web interfaces, a simple letter, a car, a mobile App, and much others that fit UX. As this dissertation is focused on a web page, the research done is focused on techniques most suitable with web pages UX process.

This section is focused on User Experience Overview, User Perception, Emotions and User Brain, Design Visual and Aesthetics, and UCD Process.

3.1.1 User Interface

UX by itself needs the UI (User Interface) to fully achieve the objectives, as they have different purposes, with UI standing for visual, and UX the entire process. The main difference is that UX focuses more on **user perception** as described in 3.2 and in **emotions** (as described in 3.3). UI is where all converts in images, colors and where visual identity takes place. A practical example about what is included in the UI part of the process, is the definition of the standards, as the colors of the buttons, the distances between elements, font faces, icons and all the style components. According to Anton Nikolov [Nik19] visual consistency means that a design is intuitive, and that users will learn faster how to use it. The first touch with the product is important to observe, as this moment indicates if what

the user pretends to do with the product is done as he thinks and if the product is easy to use. UI part of the process is described with more detail in section 3.4.

3.1.2 User-Centered Design

User-Centered Design process (UCD) as defined by Travis Lowdermilk [Low13] is a project approach that puts the user (many times the customer) at the center of design and development process, ensuring that the product will be easy to use and focus on providing a better experience for real user needs. The main goal of UCD is to make sure that the involved designers and developers know the user difficulties, and will be focused on giving a solution by including the customer in the process. The process is composed of four phases: **Research, Ideation, Design and Test**.

The main advantage of including the customer since the beginning of the process is that the most relevant issues will be discovered in the initial phases, avoiding restarting the process in a more advanced phase. Specific techniques are suitable to each phase, in 3.5 a detailed description for each phase is done in order to describe each one characteristic and present some techniques that are suited to web page UX.

Based on the information gathered by the research a simple and understandable solution for users will arise, after having a defined solution, the design will be thought and implemented as a sketch, and then a prototype will take place and finally the product. In each of these phases, tests will be done with the customer, even if the last phase is defined by product testing exclusively.

3.1.3 Design Thinking

In the wide range of human-centered subjects, design thinking is mentioned by Don Norman [Nor13] as not only a mindset, but also a design process of design focused on end-users, and is used not only by designers, but also by innovators. As a design process, this occurs before development, based on Interaction Design Foundation[Fou20b], this process has only five steps where the last consists in tests, but in some other references, there is a last step, that is called Launch, and focus on the development of the discovered solution, as in [IO20].

These are the five referred steps:

- **Empathize:** This step focuses on problem definition. Tools applied to define the problem are: user interviews, direct observation, contextual inquiry and diary studies;
- **Defining:** By the gathered information in empathize step the problem will be defined (and then the solution will be encountered based on the next phases);

- **Ideation:** The step where the ideas start to be considered as potential solutions. This phase focuses mainly on team resources and competences that will help to solve the problem;
- **Prototyping:** Consists of test solution ideas with end users;
- **Testing:** Test in order to know if the prototype is easily interpreted by the user and if it resonates with their needs.

This process can be used dynamically by revisiting some steps before, or by restarting the entire five-step process.

3.1.4 Accessibility

As defined by World Health Organization (WHO) [Org11]: Disability is part of the human condition. Almost everyone will be temporarily or permanently impaired at some point in life, and those who survive to old age will experience increasing efforts in functioning. When a new product is considered, designed and developed, accessibility is related with the very important and much of the times neglected. According to WHO disabilities report in 2011 [Org11], about 15 per cent of the population had some disability.
Still in this report is mentioned that types of disability are defined using only one aspect of disability, such as impairments – **sensory, physical, mental, intellectual** – and at other times they conflate health conditions with disability. People with chronic health conditions, communication difficulties, and other impairments may not be included in these estimates, despite encountering difficulties in everyday life..

Accessibility issues can be taken by people without physical disabilities too, according to David Benyon [Ben10] there are five ways to exclude a user to user a given product:

- **Physical** — takes too much strength to use;
- **Conceptual** — has hard-to-understand instructions;
- **Economic** — is too expensive;
- **Cultural** — users can not understand metaphors regarding product interaction;
- **Social** — on joining a group, users do not understand their social conventions.

In the web page context, Interaction Design Foundation [Fou20a] points some important accessibility guidelines (presented in 5.4) to take in account when a web page is developed.

3.1.5 Responsive Design

Even for people without disabilities, the layouts of the screens should be accessible, and have the information aligned and visible in all screen resolutions. Web browsers are used in screens of many different sizes, this way the design of the layouts should adapt to each

screen characteristics. As referred by Interaction Design Foundation [Fou20d], CSS (Cascading Style Sheet) should be applied in order to display the better version for each screen size, by using percentages in the size of the components instead of a fixed size. Besides the screen size, some parts of the page should be shown only for big screens, and not for small. One example, is the menu of the page, that in many web pages is not visible for small screens, it will only be shown when an icon is clicked. And the menu appearance will be different from the one on the big screens.

3.2 User Perception

User Experience is related with the user perceptions of the system, mainly user feelings, emotions, physical and psychological answers and behaviours making the user satisfaction the main subject of this study.

3.2.1 System

Given the subject, three factors arise: first, the **system**, second, the **user**, and third, the **context of use**. Users will interact differently according to the context, different users will react differently about a given system, and the system will behave differently by the way it is used.

3.2.2 Touchpoints

Two people can look at the same data and have two totally different readings [Zal03]. So user experience focuses in creating a global positive and consistent experience in each *touchpoint* with the user, to improve their experience. As referenced in the book Human Systems Engineering and Design [AT18], a *touchpoint* is any interaction (including encounters where there is no physical interaction) that might alter the way that a customer feels about a product, brand, business or service.

Touchpoints are important, as they are the means of contact between the company and the customer, and allows the company to gather important information to take in account in the future developments.

3.2.3 Affordances

Even in a project and development phase there are clues that indicates that a project is going well, for example when a simple task needs instructions (as labels), this indicates that the design has failed, and that the system is hard to understand. In order to understand what is easy or hard to the user some psychology subjects needs to be reviewed, for example the term *affordance*, that refers to the perception of some object, primarily those

fundamental properties that determine just how the thing could possibly be used, for example a chair affords support, and therefore affords sitting [Nor13]. When the behavior of the user is not taking the expected result in the system, the user will repeat their behavior, but the action will be ineffective, and the user became frustrated.

3.2.4 Mental Models

The action that make users try to use a product by a different way from what is supposed is based on past interactions with other similar products. This is named in psychology by **Mental Model** [Nor13]. Users create patterns in their brain, and act according to these in the first interaction with a new product.

It is because of Mental Models that it is so important to do a market analysis before a new product is created. It is also important to follow patterns from other similar products, but this is not a rule, as sometimes a user prefers to be impressed, and find a different behavior as all other similar products. For example, a car buyer can expect a different car depending on its expectations, some people prefer simpler models, and others like more sophisticated and unusual ones.

This example illustrates The paradox of technology[Nor13] once the technology offers the potential to make life easier and more enjoyable, but at the same time complexity is added, increasing difficulty and frustration. Once completely new and different, the devices become complex and difficult to operate, requiring time to adapt. [Nor13].

3.2.5 Natural Mappings

In order to reduce the adaptation time, the **Natural Mappings** concept must be considered. It is defined by taking advantage of physical analogies and cultural standards, leading to immediate understating, as well as the four constraints defined in the Design of Everyday Things book [Nor13]: physical, semantic, cultural, and logic. About the **Physical** constraints, as the name suggests, it observes the importance of the physical constitution of the object as a clue about how it operates. **Semantic** relies on the understanding of the situation, making the knowledge as the clue. **Cultural** is based on standards and stereotypes. For example, the car light above the car of the police is always blue, the stop light color is red, and so on. And the **logical** mappings, that implies logic, for example on a puzzle build, when only a place is available it is logic that the piece will belong to the remainder place.

In short, designers and developers planning a new product need to analyze the user, mainly by doing user research ([AD03]), but also be careful about context where the product is applied and analyzing other similar systems in order to take advantage of what went well

and do not fall in the same mistakes.

3.3 Emotions and User Brain

The product development is more connected with human emotions than one might initially think. As referred before, the user satisfaction is the main subject of this study, and is very related with the subject that will be presented: **Emotional Design**.

3.3.1 Emotion Theories

Emotion is specified as a feeling derived from a circumstance, mood or relationship with another person. Humans use emotions every day to make decisions based on emotional response to a given situation. Emotions help humans to realise the world.

Along the history, emotions started to be defined by Aristotle, that defined 9 main emotions, anger, friendship, fear, shame, kindness or benevolence, pity, indignation, envy, or jealousy and love.

But more complex emotions were determined after by Plutchick [Fou20c], and shows in a wheel, that represents eight primary emotions, and where the emotion in the opposite side means that is an opposite emotion, that means that a person cannot feel opposite emotions at the same time: Joy, Sadness, Acceptance, Disgust, Fear, Anger, Surprise, Anticipation.
All other emotions are derived from one or more of these eight primary emotions. For example the combination of anger and anticipation results in Aggressiveness, that is considered a complex emotion. Emotions that are represented in the center of the wheel are more intense that the one on the borders, that last ones are defined as more muted emotions.

According to Interaction Design Foundation, [Fou20c] designers should aim to create designs that increase positive emotional responses from its users, and produce designs that can tap into the emotions of its users to provide a greater level of user experience. Emotional design can transform a good product into one that users will rave about and tell everyone they know about. Two products can offer the same functions but if one can evoke a positive emotional response from its users, it will be more successful than its competition, that way, emotional design has the ability to turn users into fans.

Interesting findings about emotions:

- Intensity: is often found in posture and gestures;
- Sex: Women tend to be better at recognising emotions than men;

- Age: Older persons are better to identify emotions;
- Culture: More accuracy on the culture depends on the part of the body we are looking for.

3.3.2 Emotional Design

End-users are less receptive to products or interfaces that make them feel bad, as emotions represent a central role in the human ability to connect with the world. As mentioned by Don Norman (a widely regarded person for his expertise in the fields of design, usability engineering and cognitive science) [Nor03] there are three levels where humans emotionally connect with objects: visceral (about the first reactions to a certain product), behavioural (about our assessment of how well the product performs), and reflective(about the impact the product usage will have in our lives).

Emotional design as the name refers is based on end-user emotions, in projecting products to transmit the right emotions in the right timing of product usage, making the user feel better. This subject is tricky once if we ask end-users what they need, they will not know what they really need, as Don Norman [Nor03] refers we are not logical, we are emotional.

The future of products design is about creating engagement and commitment in a way that impact business results and measurable goals. A product needs to be persuasive, and fit the user needs. Digital transformation is about engaging end-users and motivate them to make decisions. Consumers make decisions based on emotions, so product design is used to change individual and collective behaviors.

3.3.2.1 Persuasive Design

Making users trust a given product depends on persuasion (persuasive design) as referred by Eric Schaffer [Sch09]:

"The next wave in Web site design is persuasive design, designing for persuasion, emotion, and trust. While usability is still a fundamental requirement for effective Web site design, it is no longer enough."

According to Gerald Zaltman from Harvard University [Mah03] persuasion is related with conscious level of end-users, as 95 per cent of our decisions are done at a non-conscious level, putting the focus on making things easier or more convenient to change users behavior, or make users follow new products or realities, with the focus on easier and more appealing experiences.

In a marketing point of view, the unconscious mind of the consumer is very valuable in terms of advertising, but can also be very harmful. When unconscious information about

end-users is acquired, this information can be used in a constructive way, but also in inappropriate ways.

3.3.3 Dual Process Theory of Mind

About the none conscious of the big part of human decisions, in *neuroscience* there is a theory named dual process theory of mind [Bla08] that describes two parts of human mind, the implicit mind (unconscious and outside of our awareness), and the explicit mind (the conscious part), where the first is fast, automatic, associative, impulsive, effortless, and emotional, and the second is slow, controlled, reflective, conscious, effort-full, and cool-headed.

Implicit reactions occur before they answer a report (reports are a traditional measure to evaluate users, where they are more conscious about what they will answer), when you make decisions even before you think to make decisions, it is a reaction.

3.3.4 Design to Avoid Errors and Negativity

A human error is defined by Don Norman [Nor13], as a human action that is inappropriate to system necessities, resulting in a deficit in the system or product. Having a mismatch between human competencies and technological requirements, errors will occur.

In order to make errors less costly, and work for reversible actions, Don Norman [Nor13] referred some key design principles:

- Make sure to put the knowledge required to operate the technology;
- Do not require that all the knowledge must be in the head;
- Allow for efficient operation when people have learned all the requirements;
- Guarantee that when the user is an expert person, or with a person that uses the product everyday, technological knowledge is not so detailed, saving the details for a non-expert person, or a person that is having the first contacts with the product;
- Use the power of natural and artificial constraints: physical, logical, semantic, and cultural;
- Exploit the power of forcing functions and natural mappings.

His perspective about errors describes that developers and designers should deal with error by embracing it, by seeking to understand the causes and ensuring it will not happen again. Users should be assisted in spite of punished. When the user meets an issue, then negativity occurs, a psychological effect defined as: negative bias [Lor16]. The negativity bias is defined as a tendency for people to pay more attention or give more weight and

importance to negative experiences over neutral or positive ones. Once humans pay more attention to bad things, dangerous things, a visceral survival instinct to keep them alive.

In UX context, this bias explains why a single usability issue weighs more to people when they look back on the overall experience than the numerous positive features.

Hoa Loranger [Lor16], introduces some best practices in order to prevent layout errors:

- Follow design standards;
- Match workflows to user expectations;
- Anticipate user concerns and address them;
- Write good error messages;
- Sprinkle delightful encounters;
- Test, test, and test.

3.4 Design Visual and Aesthetics

The visual impact assumes a very important role in UX, so that 5 visual principles are evidenced by nnGroup [Gor20]: Scale, Visual Hierarchy, Balance, Contrast, Gestalt.
These principles are related with characteristics of the elements that a web page can contain.

3.4.1 Scale

The first principle presented by nnGroup [Gor20], defends that the most relevant elements of a web page, should be the bigger ones, making them easier to find by the users. One example of the size importance is about the text size, if all the page has the same text size, it becomes more difficult to identify what parts of the texts are most relevant. So that the most important parts should be in a bigger font size, and details in lower size.

3.4.2 Visual Hierarchy

The second principle is related with the order by the elements appear in a web page, so that the information that the user should see in first place should be on the top of the page, avoiding the user to make scroll to find what is most important for him.
This principle explains why the most of the web pages have contacts, company address, and other secondary information on the footer or on the bottom of the page.

3.4.3 Balance

Balance refers to the correct distribution of the space between the elements, and about the disposal of the elements in an empty space. In other words refers to alignment, so that if an imaginary axis (an imaginary line that is used to organize a group of elements) crosses the middle of the page, the elements should be aligned (Symmetrical or Asymmetrical or Radial) making the page easier to read.

Symmetrical: when elements are symmetrically distributed according to the central imaginary axis;
Asymmetrical: when elements are asymmetrically distributed according to the central axis;
Radial: when elements radiate out from a central, common point in a circular direction.

3.4.4 Contrast

Contrast principle relates to the elements contrast and indicates that the ones with higher contrast are more visible than the ones with lower contrast.That way they become more visible and take a more important place on the page, and relevance to the user.

3.4.5 Gestalt

Gestalt principle refers to a set of principles as well as proximity, similarity and symmetry, in other words this principle focuses on the groups of information and defend that these blocks should be aligned, having the same space between each other if the information is related, or are more distant if the information is not related. Making this principle the one that focus on the organization of the elements.
As the “F” theory that says that the information blocks that compose a web page should be aligned at left side, so that the human eye will make an easier scan of the information. To explain the “F” theory, and still talking about the elements disposal in a web page, the ones that are related should not have breaks, so that if the elements can be displayed in the same line, the breaks without any reason should be avoided, as if a break exists, the user can assume that the elements after the break are not related with the above line ones. Referring to a break without any reason, is a reference to a break when there is still space in the previous line for the element, something like what happens in a text, when we put a paragraph purposely, but if the information is related the paragraph will insert confusion to the text sense.

Still related with the design of the interfaces, and in order to prevent poor layouts, Jon Yablonski [Jon20] presents twenty Laws of UX that represents the maxims and principles that must be taken in account 5.4.

3.5 User-Centered Design (UCD)

UCD is a process usually applied by designers focused on the user needs, is an iterative process composed by four phases (Research, Ideation, Design, Test), with the objective to create usable products. In this case some presented techniques and phases were applied to the development process.

3.5.1 Research Phase

In **research** phase context, the techniques are suited to describe the users and the problem that is intended to be resolved.

Personas Definition
A user persona as defined by Adobe [Blo20] is a representation of a particular audience segment about the product in question, normally defined in a one or two pages document as on the figure 3.1 where characteristics as behavior patterns (goals and frustrations), skills, attitudes, and background information, as a fictional person name, age, gender, location, a photo that identifies the type of person that is being described, and all the information that could make sense to the given research.

Some graphical information can be included, as in the figure 3.1 is included the Technology proficiency graphic, where is represented the proficiency in terms of IT and Internet, software, Mobile Apps, and social networks. The metrics available depends on the objective and the case in study.

Later in the UCD process, when possible solutions start to be identified, the most important information to obtain should be about goals and frustrations. The number of items about goals and frustrations is reduced in order to direct the focus to the essential user pains.

Xtensio [xte20] is a web tool that provides personas templates, some examples of already created personas, and allows a persona to be created based on a selected template.

Surveys
Surveys are tools that can reach many people in a short amount of time, and do not require the physical presence of the participants. There is not a defined step by step method about how to create a survey, there are no specific questions to do, only questions that will provide useful information by the given answers, according to Travis Lowdermilk [Low13], having clear, concise, and impartial questions, as well as selecting the right group of people, is the key to a successful survey. Is not completely certain that the answers correspond to the reality, that way this method is not so trusted as the user interviews in terms of gathering users answers. Examples of tools to create and analyzing surveys are: SurveyMonkey [Sur20] and Google Forms[Goo20].

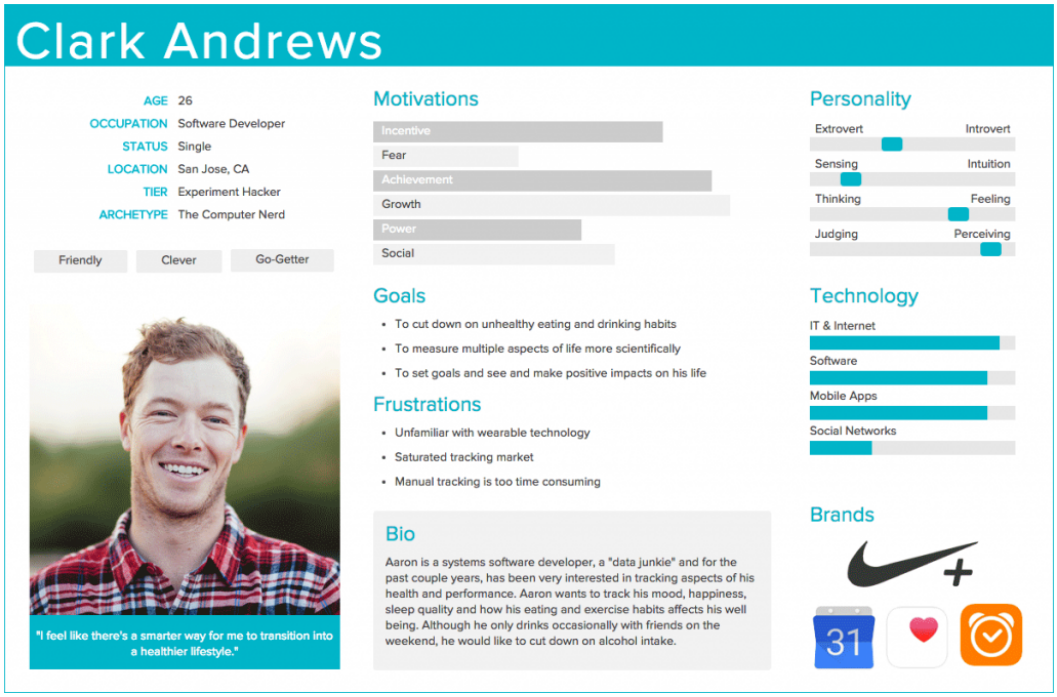


Figure 3.1: Example of a persona definition by Adobe [Blo20].

User Interviews

Is defined as a one to one session where a member of the UX team and a possible user are present, and where the answers are recorded by the UX team member. Consists in an informal conversation where the UX team member will conduct and going through some pre-defined questions, in order to make the user talk about the convenient topics, mainly about goals and pains about a given product. This type of research will give UX Team qualitative self-reported data, once responses are gathered, and at the same time will be possible to see user reactions and emotions. User interviews can be useful at the beginning of the project, when a new product is still not defined, or completely defined.

User interviews can be seen as a source of information to define Personas 3.5.1, as Surveys 3.5.1 have the same objective, by helping to decide what project features will be included, or after usability testing to add verbal self-reported data to the actions and behaviors observed by testing the product.

Before the interview:

- The goals and objectives should be defined;
- The participants recruited;
- The location defined;
- And the questions prepared.

Issues	Importance Ranking		
Name	**Users**	**Organization**	**Total Score**
Customer-service quality complaints	5	5	**10**
Search-results quality issues	5	3	**8**
Comparing stereos is difficult	2	5	**7**
Sharing feature is not being used enough	0	5	**5**
Few people listen to the podcast	0	4	**4**

Figure 3.2: Scenario importance definition by Susan Farrell [Far17].

Scenarios

According to Travis Lowdermilk [Low13], scenarios are brief stories focused on a persona, and describes how this persona as a user will interact with the product, to complete a particular goal or task.

To describe a scenario, the persona, the motivator, the intent, the action, and the resolution should be described.

According to Susan Farrell [Far17], a Seven-step method defined:

- Determine the most important user tasks;
- Discover which system aspects are of most concern;
- Group items from 1 and 5, then sort issues by importance to users and organization (as shown in the figure 3.2);
- For each top issue, condense the information into a problem statement;
- For each problem statement, list research goals;
- For each research goal, list participant activities and behaviors;
- For each group of goals, write user scenarios.

Costumer Journey

According to Alan Pennington [Pen16] a customer journey is defined by a map that is defined in a structured way in order to understand the user needs and expectations at each stage of their experience with the product, by recording the levels of empathy step by step.

This map is composed by five stages (Awareness, Consideration, Purchase, On-boarding and Advocacy), visible in the top of the map represented on the figure 3.3, and working as a ruler to all the topics on each row (User Actions, Touch-Points, Emotions, Pain points, and possible solutions). In order to populate that last ones, and still according to Alan Pennington [Pen16], some questions can be done, from where the map information will be gathered:

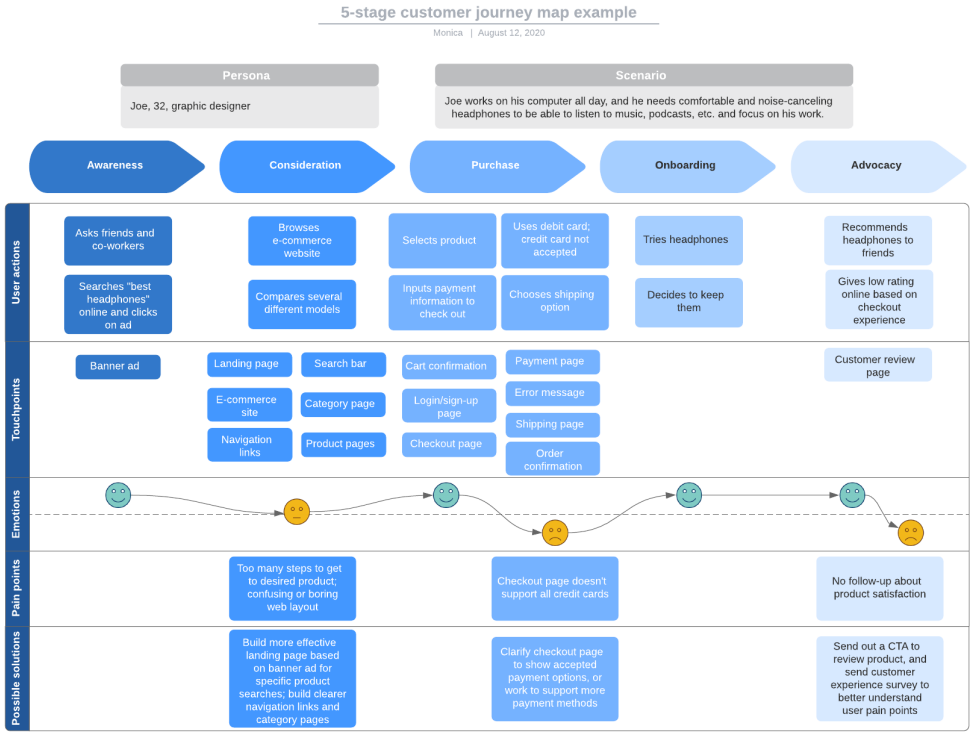


Figure 3.3: Example of a customer journey map by Lucidchart [luc20].

- What is the user action?
- What is the user thinking or feeling?
- What is the customer touch point with the company that provides the product?
- What the user need to change about this step?
- How or why this change will be done by the project team?

By using this map, it is possible to:

- Gather the key interactions;
- Better understanding of the user;
- Identify the gaps in customer experience;
- Prioritize where and how to focus on product changes;
- Identify different experiences for different users / groups of users / segments.

Figure 3.4: Example of person using card sort technique [usa20].

Information Architecture (IA) According to information architecture institute [iai20] IA "is a practice of deciding how to arrange the parts of something to be understandable", making information clear and organized to be better understood and found by the user, and avoiding user to lose time.
According to Adobe [Nic17a], IA is normally practiced in defining the navigation of the websites, but can be seen in many other situations where it is needed to relate pieces of information.
Before start building an IA diagram, the essential information should be defined. In order to organize the collected pieces of information, **Card Sort** [usa20] is a usual technique that allows to group the information pieces in categories as shown in the figure 4.8, and is practiced by members of the UX Team.

According to Usability guidelines about Card Sort [usa20] this technique can be used to:

- Build the structure of a website;
- Decide what to put in some page, like homepage;
- What is relevant for users, and should be more visible, or quick to access;
- Label categories and navigation.

After the information is grouped in categories usually a test is done with users in order to understand if the information is easy to find in the categories defined by the UX Team,

this test is called **Tree Test** [Whi17], and consists of challenge the user to find some subject inside the categories and sub-categories until he finds or not a given subject or theme.

Tree Test helps to:

- Define the key website goals and user tasks;
- Discover Potential problem areas.

3.5.2 Ideation Phase

In **Ideation Phase** the applied techniques are related with possibility exploration, and idea generation, where the UX Team will discuss the information gathered in the research phase, and convert it in **prototypes** (according to Nick Babich [Nic17b] a prototype can be almost anything, from a series of paper sketches representing the different screens or states of an app to a fully-functional, pixel-perfect app), usually paper prototypes comes first, once these will provide a less attachment between the team and the designs, and will be easier to replace or re-design new ideas that occur after the first prototype development.

Based on Jerry Cao [Cao16] article, one of the main advantages of paper prototyping is the rapid iteration, these let teams to create and throw away multiple versions without wasting many times.
Prototyping in paper is less expensive and increases the creativity, as well as increases team building.

3.5.3 Design Phase

In **Design Phase** and after the ideas are converted in prototypes, the design part takes place, and the creative process starts by implementing most real prototypes (High-fidelity prototyping [Nic17b]) than the ones referred in the ideation phase (the paper ones) and testing them with users until find an acceptable version of the prototype. According to Travis Lowdermilk [Low13] the latest phase (Ideation) consists in adding value to the product, and Design phase makes the product delightful and the combination of both will give users better experiences by using the product. Still, according to Travis Lowdermilk [Low13] creativity can be fostered by exploring other similar products and patterns to apply in the new product. This phase is the one where the User perception 3.2 is most valuable, and taken in account, as well as Visual 3.4.

3.5.4 Test Phase

The last phase, **Test Phase**, is the one where feedback is gathered, even if in the previous prototypes tests were done with clients.

According to Travis Lowdermilk, [Low13] testing is not give the product to the client in order to make tests. A script can be a useful tool to go conduct tests, and avoid distractions, according to him a test script should be composed by:

- Introduction (Introduce the users what information will be used and what the collected data will be used for);
- Reassurance (In order to avoid the users to be nervous, should be referred that they are helping in a product test, and not their skills using the product);
- Testing Guidelines (Where the topics will be pre-prepared and where should be defined if the users can do questions during the test, and how many times do they have to complete);
- Tasks (Where tasks are defined as well as the metrics that are planned to measure);
- Conclusion (In the end of the tests some time should be saved to make some questions, and share the observations collected during the test with the user);
- Thanks (Thank user for their time, and sometimes is appropriate to give a small gift in appreciation of the help).

Some other aspects are referred to by Travis Lowdermilk [Low13] as useful materials, like a Stopwatch, Notepad and Camera or Audio Recording. About test environment, this should complement the study, and should avoid distractions and be comfortable (chair, lighting, network connectivity and room temperature), a natural environment for the user will create realistic results. In the end of the tests, the results are quantified and the comments organized in order to find decisions to improve the product. This technique of observing users while they test products is called in the specialized bibliographies as **Usability Testing** [Fou18]. When tests are applied in terms of user preference, for example when two versions of the product, or two prototypes are presented to the user and the user just needs to chose what he prefers, this as referred by Jakob Nielsen are **A/B tests** [Nie20b] or **Preference Tests** [Hub20].

In terms of test, many times heuristics are applied. Heuristics are defined as a list of characteristics that a webpage or mobile application should have to be accepted, and this list of topics should be evaluated by a group of users, according to Jakob Nielsen [Nie20a], a group of five persons are enough to find the principal problems of the application, as revealed by his study about cost benefits of the number of application testers, as shown in the graphic represented in the figure 3.5.

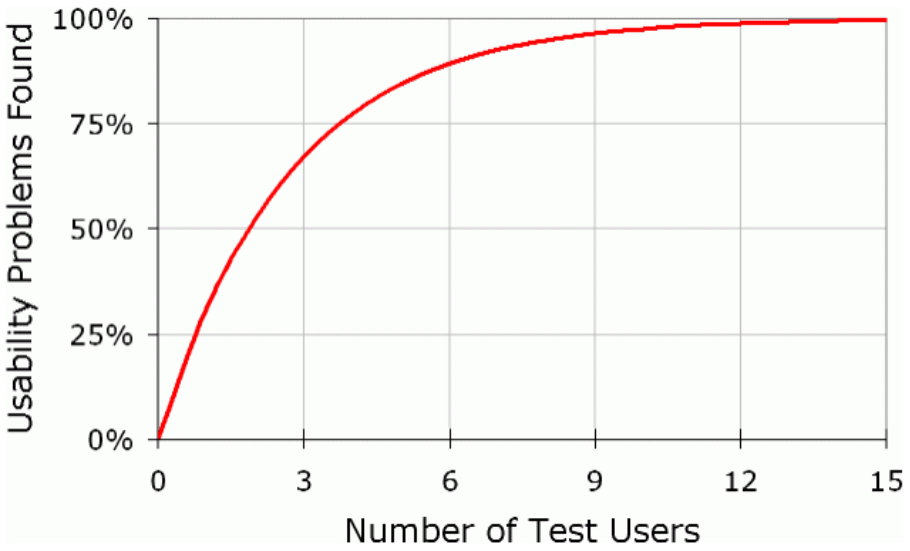


Figure 3.5: Graph about Jakob Nielsen study about the minimum number of users to find the most important flaws [Nie00].

3.6 Conclusions

The presented theories about user experience were reviewed and applied during the development process of the platform, being this phase focused on UCD design process. After the development of the first version of the platform, as described in the next chapter, in the ideation phase, the Card Sort technique 4.5.2 was applied with the objective to gather information about new features presented by users allowing to identify which ones were important to apply in the next version.

Chapter 4

Metrics and System Proposal

4.1 Tasks

4.1.1 Introduction

In order to complete this work, various tasks in different contexts have been taken into account. The first was the analysis of the state of the art about information gathering and platforms based on *OSINT* systems. Then the formulas and metrics demonstrated in Ivan dissertation [Fer16] were analyzed, as described in section 4.2.
Before the development phase, user experience has been studied, and described in chapter 3. And after this study a new platform started to be developed. Two versions were developed. After the development of the first version, the users filled a form by providing his personal opinion, that were analyzed, and new features arose and were implemented in the second version of the platform. That was followed by a new phase of tests and feedback to gather from several users, providing feedback for future work.

This section will describe the presented **tasks**, that were proposed to be completed. The **formulas** applied by the last iteration of this project, and the improvements in this topic. The **Tools** used to develop the platform and the architecture of the platform will be described. And the **Evaluation** of the two prototype versions.

4.1.2 Initial work schedule

The tasks initially defined, and the respective duration, were referred in the master dissertation proposal, presented here:

- **Task 1** Contextualize with the problem at hands and with the objectives of this project, as well as with the technologies involved and with the previously developed prototype [Fer16]. Revision of the specialized literature and related works. Definition of the method for evaluation of the system (two months);
- **Task 2** Detailed analysis of the formulas devised for the previous iteration of the system and study of the impact of tweaking weights associated with the metrics used in the formulas (one month);
- **Task 3** Proposal of new metrics for the classification of posts and twitters, followed by their integration in a fork of the system (one month);

- **Task 4** Assessment of the usefulness of the new metrics in the classification effectiveness and comparison with the previous version of the system (one month);
- **Task 5** Improvement of the way information is presented to the user of the system (one month);
- **Task 6** Evaluation of the system (one month);
- **Task 7** Writing of the master's dissertation, technical documentation and a conference paper (three months, eventually distributed and interleaved with the time periods of other tasks).

4.1.3 Task Identification, schedule and Work breakdown

The tasks developed throughout this dissertation were related with six phases, starting by the state of the art, then the metrics analysis, the User Experience research, the platform development, the platform evaluation, and the dissertation writing, as represented in the figure 4.1.

In the **state-of-the-art** phase, papers and bibliographies were considered in order to better understand the thematic, that was reflected in the second chapter of the dissertation (two months).

In the **user experience research** phase, the study of user experience techniques led to chapter three writing (one month).

In the **metrics analysis** phase, the metrics and formulas presented by Ivan [Fer16] were analyzed, a new database schema was planned, and the metrics were applied to a simple prototype, that started to be developed, and presented in two versions (one month).

Based on the collected information and ideas, the **development phase** started, where the first prototype was developed, a simple version, with a search page only, this version were very similar to the platform presented by Ivan [Fer16], where various topics were presented in a combo-box and by clicking a search button the posts about the selected subject were displayed by relevance. But this page by itself did not meet all the requirements. So this page was presented to some users in order to collect feedback (two months).

Based on the collected feedback about the first version a **new version** was developed, where the topics started to be configurable, allowing to check posts from various subjects at the time, as well as some other configurations, as the configuration of the number of minutes to see the posts, and the sort by relevance or date, making possible to see only the last tweets in a configurable number of minutes, or see first the most relevant ones in a wider range of time. Some monitors were presented in order to show the number of posts by various ranges of time and the relevance by color (two months).

After the new version was completed, a new **evaluation** was performed with users, gathering feedback, parts of this feedback were implemented, and others were marked as future work (1 month).

The last phase, the **dissertation writing**, it was not exactly the last task, it was being written over the various phases, with chapters 1, 4 and 5 being the last to be written (1 month).

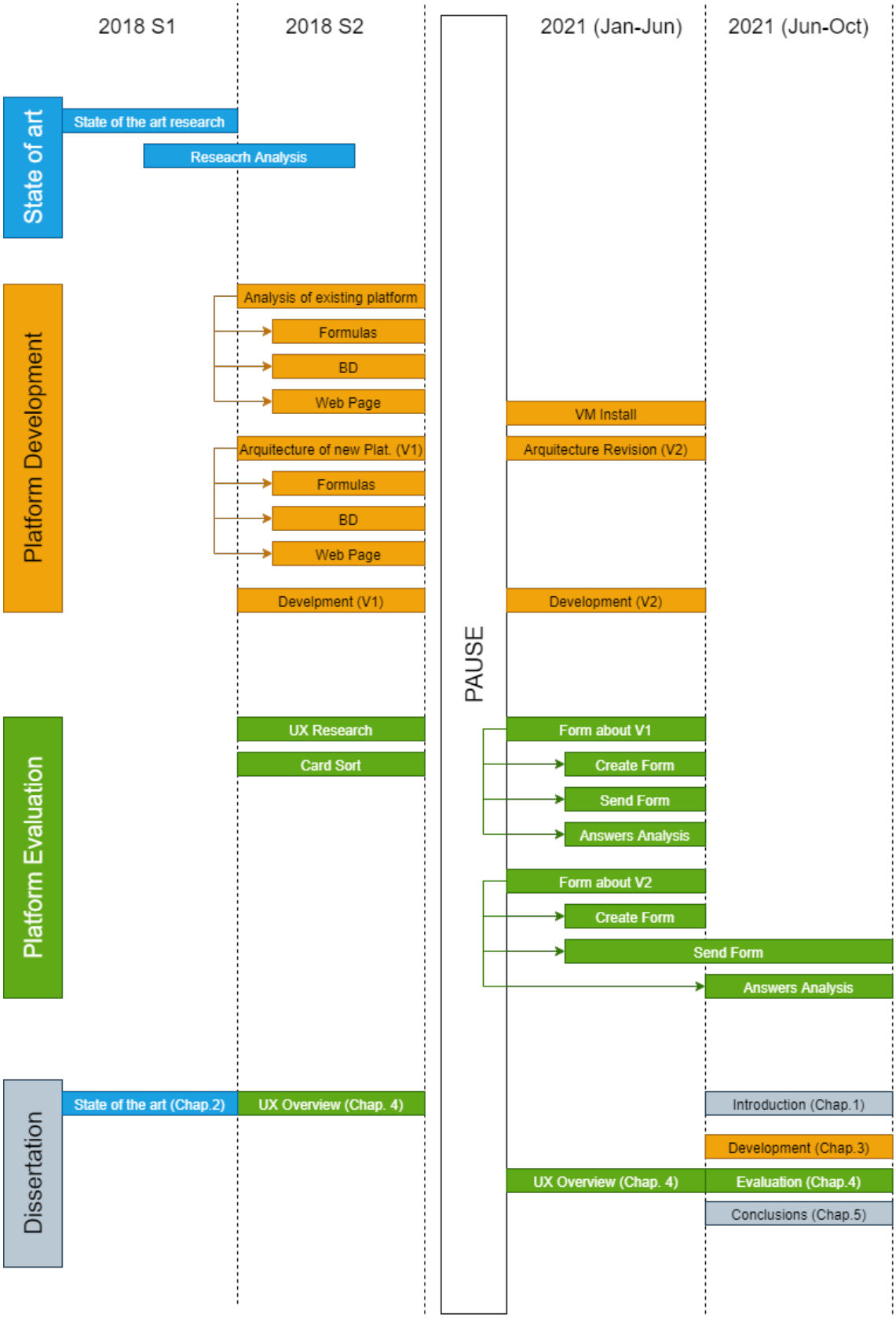


Figure 4.1: Graphical representation of the planning of the tasks envisioned for the project described in this dissertation.

4.2 Formulas

With this dissertation being the continuation of a set of formulas and metrics, in order to calculate tweets relevance, this section will describe the formulas applied in the last iteration, and presented by Ivan in his dissertation [Fer16], the post relevance formula, and the user relevance formula described in 4.2.3, and finally the contribution of this work on the word relevance about *InfoTermo*, in terms of relevance of the words by domain described in 4.2.1.

4.2.1 Word Relevance

Due to the wide range of topics available, a library of words was created in order to save all the words in a given domain (referred as subject in the database schema), calculating or recalculating the word domain relevance $P\{w|domain\}$ each time a new post is processed and inserted in the word table in the database 4.3, this way all the post words, free of stopwords (words that have no relevance, and that only make sense to connect the words of the text, and that alone do not make sense) are saved or updated (in the case that the word is already present for the post domain), as represented in listing 2.

Each time a new word is inserted or updated, two important variables are incremented $nPostsOcurred$ and $nPostsEvaluated$. Being the $nPostsOcurred$ the number of occurrences the word occurs in the posts, and the $nPostsOcurred$ the number of posts evaluated. So the word frequency in the domain is given by:

$$P\{w|domain\} = \frac{nPostsOcurred}{nPostsEvaluated}; \quad (4.1)$$

By having $P\{w|domain\}$ the *InfoTermo* formula will be applied in order to indicate the word relevance in the corpus of the text:

$$InfoTermo(w|domain) = \sqrt{|w|} * log_2\Big(\frac{P\{w|domain\}}{P\{w\}}\Big). \quad (4.2)$$

where w is the given word, and $|w|$ is the number of characters, $P\{w\}$ is the probability of w estimated in a representative *corpus* of the language, and $P\{w|domain\}$ is the probability of the same word but in a subset of text relative to a given domain, as referred to formula 4.1.

Therefore, the more words that are inserted, and the more posts are analyzed, the more trustworthy will the relevance of the domain words be. Listings 1 and 2 show the application of these formulas.

Listing 1 Post Relevance

```
//Return the map of words frequency in the text.
function postRelevance(string, numLikes, numRetweets) {
    var words = stopwords.removeStopWords(string);
    var freqMap = {};
    var uniqueWordsArray = [];
    var infoTermoSum=0;

    words.forEach(function(w) {
        if (!freqMap[w]) {
            freqMap[w] = 0;
        }
        freqMap[w] += 1;

        if(!uniqueWordsArray.includes(w))
            uniqueWordsArray.push(w);
    });

  words.forEach(function(w) {
    infoTermoSum+=InfoTermo(w, WordProbability(freqMap[w],words.length));
  });

  var nUniqueWords = uniqueWordsArray.length;
  var postRelevance = (1/nUniqueWords)*infoTermoSum;

  var finalRelevance=
  ↪  0.5+(Math.atan(Math.sqrt((1+numLikes)*(1+numRetweets))*postRelevance)/Math.PI);

  return finalRelevance;

}
```

Listing 2 Domain Relevance

```
30 function postRelevanceDomain(tweet, connection) {
31 var texto=stopwords.removeStopWords(tweet.text).toString();
32     if(texto!=""){
33 
34             var freqMap = {};
35 
36                 var palavras = "'"+texto.split(/[ ,]+/).join("', '")+"'";
37 
38                         $query = "select * from word where word.name in ("
                            ↪  + palavras +")";
39 
40                         connection.query($query, function(err, rows,
                            ↪  fields) {
41 
42                         if (err) {
43                             console.log("WORDS -> An error ocurred
                                ↪  performing the query.");
44                             return;
45                         }
46 
47                         for (var item in rows) {
48 
49                             if(rows[item].nPostsEvaluated>0)
50                             freqMap[rows[item].name] =
                                ↪  rows[item].nPostsOcurred/rows[item].nPostsEvaluated;
51 
52                         }
53                         setTweetRelevance(tweet, freqMap, connection);
54                         });
55     }
56 
57 }
```

Listing 3 User Relevance Query

```
58 SELECT 1/ COUNT(eval.evaluation)*SUM(eval.evaluation) as userRelevance,
   ↪  SUM(eval.evaluation) as userEval,
59 COUNT(eval.evaluation) as numPosts,
60 user.Name as userName, tweet.dateCreated >= NOW() - INTERVAL
   ↪  '+searchMinutes+' MINUTE as include, tweet.*, user.*, subject.*
61
62 FROM tweet, user, subject
63
64 JOIN (SELECT  tweet2.evaluation as "evaluation", tweet2.userId as
   ↪  "userId", tweet2.SubjectId as "SubjectId"  from tweet as tweet2 where
   ↪  tweet2.evaluation>0 order by tweet2.evaluation DESC) eval
65
66 where tweet.userId=user.Id and tweet.SubjectId in  ('+subjectsArray+') and
   ↪  tweet.tweetLanguage="'+selectedLanguage+'" AND
   ↪  eval.userId=tweet.userId AND eval.SubjectId=tweet.SubjectId AND
   ↪  tweet.evaluation>0 and subject.Id=tweet.SubjectId
67
68 GROUP BY tweet.Id, user.Id
69 ORDER BY userRelevance DESC
```

4.2.2 Tweet Relevance

Having the post words relevance calculated and saved in the corresponding database table the two formulas demonstrated by Ivan [Fer16] can be applied in order to get the final tweet relevance, *RelevF*, including the number of likes and retweets, but in reality this number is in the most of the time zero, as the posts are collected in *real-time*.

$$Relev(post) = \frac{1}{n}\sum_{i=1}^{n} InfoTermo(w_i|domain). \quad (4.3)$$

$$RelevF(post) = \frac{1}{2} + \frac{arctan\left(\sqrt{(1+\#Likes)(1+\#Retweets)} * Relev(post)\right)}{\pi}. \quad (4.4)$$

4.2.3 User Relevance

The relevance of a user is given by formula 4.5 that was presented by Ivan [Fer16], and has in consideration the texts published, in this case the posts. So the average relevance of the last m posts $(p_1, p_2, ..., p_m)$ related to the domain the post inserted):

$$RevUser(u) = \frac{1}{m}\sum_{p_i \in u} RelevF(p_i). \quad (4.5)$$

User relevance is calculated in real-time, and is applied in the webpage process, and where the presented query in 3 is applied.

4.3 Tools

The presented prototype is available in a Windows 10 Azure Virtual Machine, where a Node JS Application is installed, in order to run the web page.
The web page refers to a Node Web Server, with access to database queries, that are responsible for gathering the tweets displayed at the platform. The referred database is filled by a *Stream* that is called by a Node JS process, and collects posts through the *Twitter* API, as can be consulted in 4.4.1.

4.3.1 Node JS, Javascript and CSS

The node JS [Fou21] platform is an asynchronous event-driven JavaScript runtime designed to build easy-to-scale network applications. Where it is easy to make a web-page available, and where the applications can deal with server-side JavaScript and Client-Side JavaScript.
The developed prototype has two views, where the server side processed information about posts is passed, and where the server side processes the structure of HTML that will be displayed to the user.
On the client side two themes are passed, where two CSS (Cascading Style Sheets) are disposed to be dynamically selected, if the user selects the dark mode the dark mode style-sheet will be selected, otherwise the normal style-sheet will be the base of the web page style.

4.3.2 PHP MyAdmin

PHP MyAdmin [Adm21] was the selected database engine to administrate the SQL based platform information, as it is free and one of the most applied ones with node JS applications, and is composed of a web database editor where most of the administration features are supported. In 4.4.2 the database model created for the prototype is shown.

4.3.3 Azure Platform

A Windows 10 virtual machine was installed in Azure Platform [Mic21] in order to make the web page available for the users, mostly in order to provide the platform for users for testing purposes. In the virtual machine the XAMPP [XAM21] server was installed, an Apache distribution that comes with PHP MyAdmin, and Node JS.

4.3.4 Twitter API

Twitter has various APIs available for developers, in order to make tweets and other related data available. In this case the Stream API and the Search API were used, the first for the second version of the prototype developed in this dissertation, and the second for the first and simpler version. About the Stream API, some important parameters can be passed, as the language and the subjects to search. Tweets are returned in a JavaScript

Object Notation (JSON) structure with all the information about the tweet, the user, and the media included in the post, and can be seen in an example available in Appendix 5.4.

4.3.5 Bootstrap

In order to get a most flexible and responsive layout, the Bootstrap library [Boo21] was implemented. This is a *CSS* and *Javacript* based library, that makes available CSS classes and JavaScript functions in order to get specific components to work and adapt to various screen sizes. So by using the indicated HTML notation for a given component, this component will have a given behavior and adaptation to the screen as described in the Bootstrap Documentation [Boo21], avoiding the creation of unnecessary scripts and CSS styles.

4.4 Platform

The presented platform is based on three processes, the first one that gather the posts, where the **stream** Twitter API is called, and collecting all the tweets except the ones without useful information, as the retweets (where the tweet text starts by "RT"), the tweets with less than six words, the ones composed of links or useless information or *stopwords*. The posts that pass these validations are inserted in the database table "tweet", as well as the user information, in "user".
The second process is the one that applies the relevance formulas, the one responsible for the *tweet* **classification**, where all the present tweets in the database and that have not yet been evaluated will be classified (this means that the tweet evaluation attribute will be filled by this process).
The third process is the one that makes the **web page** available in the 49153 port. On the *web page,* the information comes from the database directly, and only the evaluated posts are shown. All this processes run infinitely.

4.4.1 Architecture

The three processes work together to make the web page available with tweets to show to the users.
As shown in the figure **??** the Stream process is the only one with access to the Twitter API, where the posts are gathered. This API returns a lot of information about the tweet, as some simple metrics, as the number of followers of the user, and the number of likes of the tweet. Besides this last one is not very relevant, once the tweet is collected when it is posted, so at this time has not likes yet.
Besides the API returns much information about the post, only the most important fields for this platform are saved to the database, in "tweet" and "user" tables, as shown in section 4.4.2.

As referred before, the three processes that compose the application are actually three applications developed in Node JS, composed for various files, each one with each own responsibility as represented in the figure 4.2, where the file tree is presented.

In the tree where files are represented in listing 4, the nodes **CLASSIFICATION**, **STREAM** and **WEBPAGE**, represents the tree processes.

Each one of them has a `node_modules` folder inside, the one that contains the node JS installed modules, that can be configured folder by folder, that means that the modules are exclusive for each one of them, one module that is needed in two different app folders, needs to be installed in both folder paths. All the folders have an `app.js file`, that is the one called by node JS, the main one that will call secondary files each time is needed. Two files that are present in all the directories, *package-lock.json* and *package.json*, are relative to node JS.

In the **CLASSIFICATION** folder, there is the *app.js* file, the one that is called by node JS, the primary one, and the *stopwords.js*, this last one corresponds to a class that receives a text and then removes all the stopwords, returning only the words that represents useful information, as represented in listing 5.
The stopwords removal algorithm, two node JS modules, *stopword* [NPM21b] and natural [NPM21a]. The first one responsible for checking stopwords in a given language, in this case English, and the second to *tokenize* natural language words.

In the classification process a query is done gathering the posts that has not yet been classified (where the evaluation field is equal to zero), and then classifies one by one according the words that contains, all is done by subject.
In order to classify the *tweets* the applied formulas are the ones presented in 4.2.2.

The **STREAM** main app, app.js is responsible for gathering the information from Twitter API and save the tweets and respective users in database, as well as register all the words and update their occurrence in the *word* table. As this process establishes communication with Twitter API, the node twitter module[NPM21c] is used. *MYSql* and FS (FileSystem) modules are also used in this process. FS is used to save logs to text (txt) files, each tweet that is collected is inserted in a text file, this feature is disabled in the installed version (in order to save resources).
The config.js file contains the keys to access Twitter API, as shown in Listing 6.

In the **WEBPAGE** is where the app.js gets all the tweets by the specified page filters, directly from the database.
A sub-folder named ”views” has two .ejs files, the error, and the home, the error is the one that is called when some error occurs in the application. The home is the main view, the one that is applied in the search page, and where the tweets are displayed. By using

Listing 4 Node JS Files

```
+---CLASSIFICATION
|   |   app.js
|   |   package-lock.json
|   |   package.json
|   |   stopwords.js
|   |   {
|   |
|   \---node_modules
+---STREAM
|   |   app.js
|   |   config.js
|   |   package-lock.json
|   |   package.json
|   |   rel_dom.js
|   |   stopwords.js
|   |
|   +---logs
|   |       06-19-2021.txt
|   |       06-20-2021.txt
|   |       06-22-2021.txt
|   |
|   \---node_modules
\---WEBPAGE
    |   app.js
    |   index.html
    |   package-lock.json
    |   package.json
    |
    +---css
    |       app.css
    |       app_dark.css
    |       dark.css
    |
    +---node_modules
    \---views
            error.ejs
            home.ejs
```

Listing 5 Stopwords

```
module.exports = {
removeStopWords: function(sentence) {

        const withSW = sentence ? sentence.split(' ') : '';
        if(withSW!=''){
            const noSW= sw.removeStopwords(withSW, sw.en);

            var tokenizer = new natural.WordTokenizer();
            var WordsNoLinks=[];
            for(nword in noSW){
                var wordNoPunctuation =
                ↪ noSW[nword].replace(/[.,\/!?$%\^&\*;:'{}=\-_`~()…"]/g,"");
                ↪ //Remove punctuation
                    wordNoPunctuation = wordNoPunctuation.toLowerCase();
                    ↪ //convert to lowercase
                var reNumbers =/\d/g; //Validate numbers

                if(!reNumbers.test(wordNoPunctuation))
                if(!wordNoPunctuation.startsWith("http") &&
                ↪ !wordNoPunctuation.startsWith("@") &&
                ↪ !wordNoPunctuation.startsWith("#") &&
                ↪ wordNoPunctuation.length>6){
                  WordsNoLinks.push(wordNoPunctuation);
                }
            }
                 return tokenizer.tokenize(WordsNoLinks.toString());

        }
        else{
            return [];
        }

}
};
```

Listing 6 File - config.js (Twitter Configurations)

```
module.exports = {
          consumer_key: 'HgNhGWeQxFrAx8ZpijuIYskY8...',
          consumer_secret:
          ↪ '9kEfewlzycDhmFCEKM16KTw5tlQYsLehPXKwV8h1WW0Tkdz...',
          access_token_key:
          ↪ '838135690936352770-dPIgg9AisgGFGBYesCSLo15RRoze...',
          access_token_secret:
          ↪ 'lmZDvZWpyLheU8x2L4eARSPYshF4Yk0MPZeIhGj38z...'
}
```

home.ejs, the main file app.js, sends the data already in the format that needs to be displayed, so the tweets are pre-processed server side, and data is sent to the views as HTML. The other specific folder is the CSS, the one that contains the main theme *app.css*, and the dark theme, *dark.css*, as well as all the necessary adaptations for dark theme to run as the main theme but only with different colors, in *app_dark.js*.

MySql and cookie node module was installed in order to get information about the cookies in the user browser, the user configurations are saved in cookies. Cookie values are sent to the server side each time the user reloads the page, or changes the configurations.

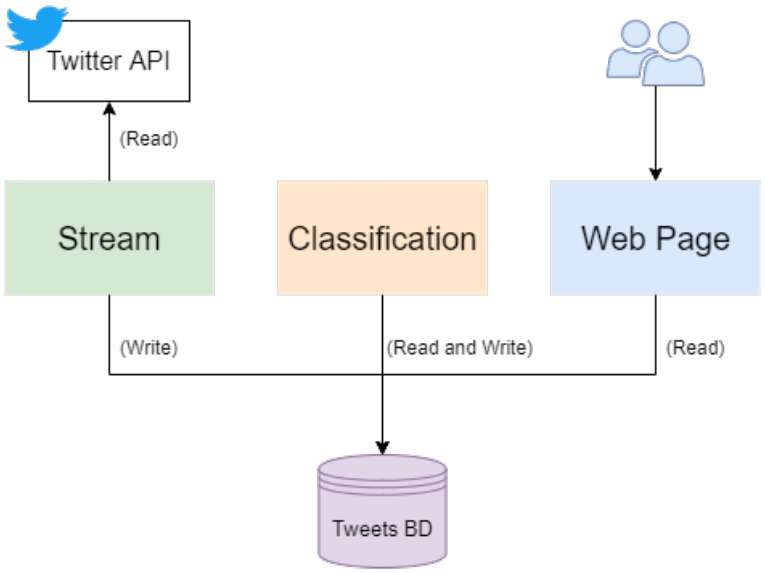


Figure 4.2: Platform Architecture representation.

Within the tweet classification **process** there is a task that **deletes tweets** in order to the database do not grow exponentially, and in order to improve the web page performance.

4.4.2 Database Modeling

The database model is composed of four tables, as shown in figure 4.3. The main table, *tweet*, that gathers information about tweets, the user that is connected with tweet by a foreign key, allowing to save user information about who inserted the tweet. The *subject table* refers to the table that saves the subject of the tweet has been gathered from Twitter API. A selection of the content of this table will show to the users the subjects that they have available, otherwise, just subjects inserted in this table will be streamed by the stream process. The word table is the one that saves all the words that compose a tweet, word table has no reference for tweets, as this table is seen as a library of words and the number of times they are shown for a given subject. So this table only has references to the subject. Each subject has an associated color, that will visually identify the name of the subject in the web page tweets list.

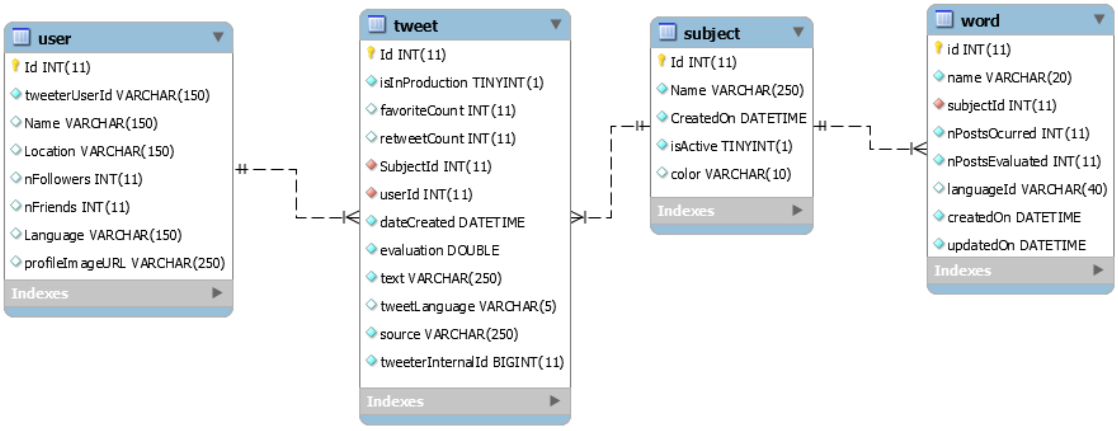


Figure 4.3: Data Model tables representation.

4.4.3 Web Page

The web page counted with two versions, where the first one were a simple search page, with relevance calculation, and the second one by a most complex platform, with configurations and filters.

4.4.3.1 Version 1

The web page development started by a simple prototype, where only a simple search by subject and language were allowed. This first version were based in the Twitter search API [Twi21b] instead of the stream API [Twi21a] used in the second version. This version is not compatible with the three processes presented in the Architecture description of the platform, in 4.4.1, as in this version all the tasks are part of the same process, classification, stream and providing the web page. The tweets were classified the same way in the second version, the difference of the two versions were in the web page layout, functionalities, and in the way the platform gather the information. In the first version the information was gathered when the user searches by a subject (by selecting in a combobox the available subjects, and when the search button were pressed, a defined number of tweets were collected to the database).

As defined in the search API [Twi21b] the search results were filtered by an optional parameter (result_type), giving the most recent results, by passing “recent” word, the most popular by passing the word “popular”, or a mix of the two filters, by passing the word “mixed”. In this case the recent filter was applied, but a problem occurred, if two users, or the same user searched the same subject at the same time, or in short time, some of the same tweets can be returned, so the algorithm needed to take in account this situation in order to do not insert repeated tweets at the database. The layout of this version was based in Bootstrap components and in an example Template.

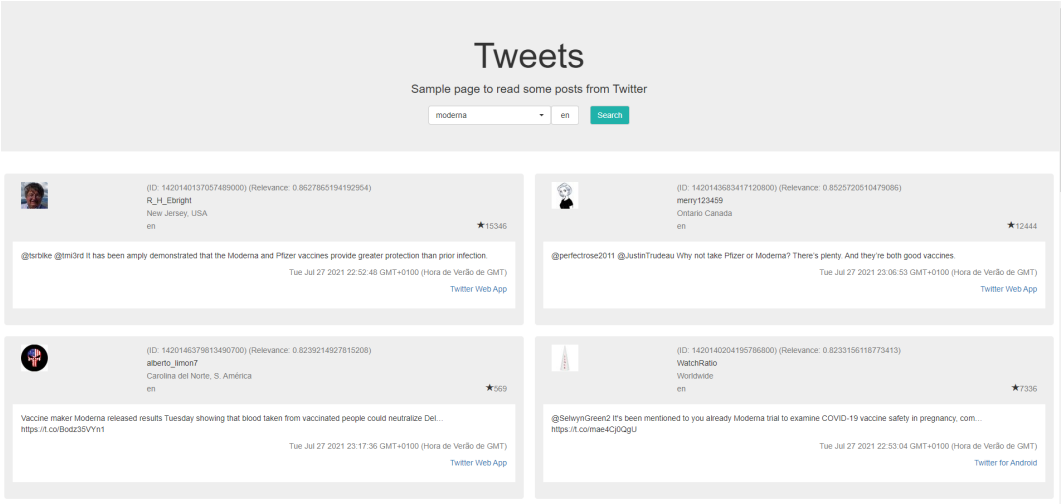


Figure 4.4: Web Page — first version.

4.4.3.2 Version 2

The second version is an improvement of the first version, based on user testing inputs, and user ideas.
This one is based in the three processes model presented in the architecture, in 4.4.1, where the stream gathers the information, the classification process, classifies the process, and here the web page is a single process, so only calls to the database are done by the webpage in this version, instead of consulting Twitter API every time a search occurs. Other improvements were inserted as well, as the monitors in the top of the page that allow the users to visualize in temporal windows how many tweets by relevance were collected. Another important improvement respects to the configuration menu, represented in figure 4.5 where the clear mode or dark mode can be selected, the number of minutes ago to show the tweets, the subjects that the user wants to follow, an input box where a new subject can be inserted, and the possibility to order the list tweets by date as shown in figure 4.7 or relevance, as shown in the figure 4.7. The difference between these two figures is that the one filtered by relevance is showing less irrelevant tweets (marked with red and orange).
The relevance colors depend on the evaluation given to the tweets, and according to the evaluation a color is applied, these colors and evaluation ranges can be consulted in the listing 7.
The layout is different from the latest version, once the space that each tweet occupies is lower than in the last version, and some information were not considered relevant by some users and has been removed. New tweets were highlighted with green and a label that indicates that is a new tweet (tweets are considered new tweets if they were gathered in the last five minutes).

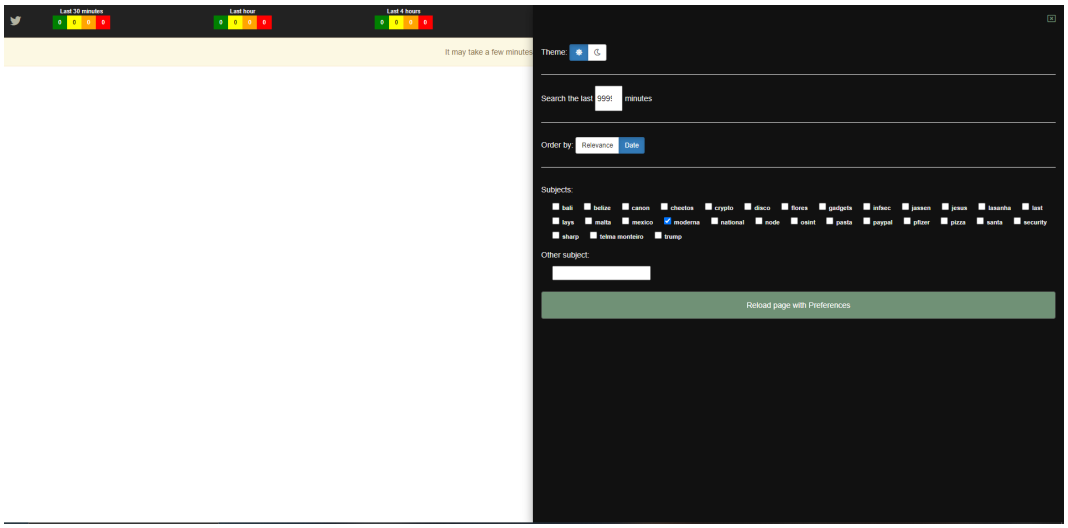


Figure 4.5: Web Page — second version — configurations menu.

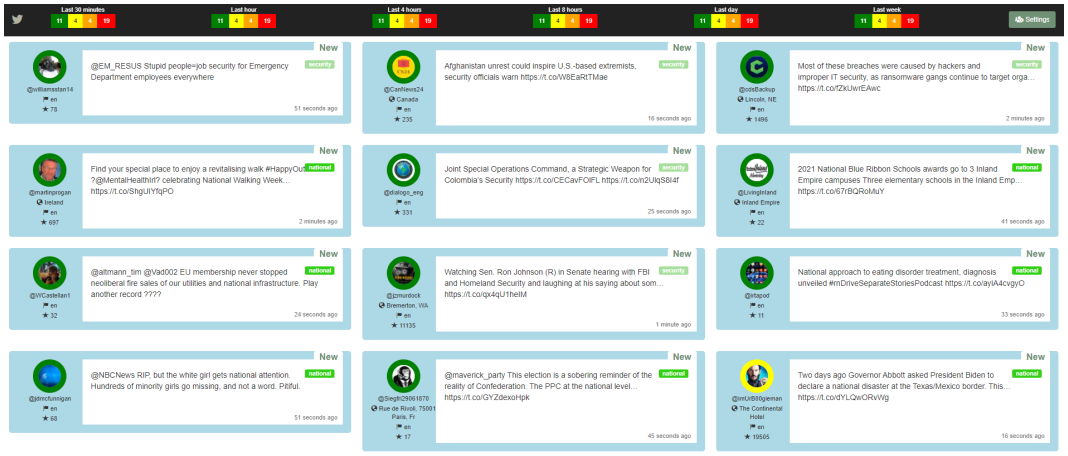


Figure 4.6: Web Page — second version — ordered by relevance.

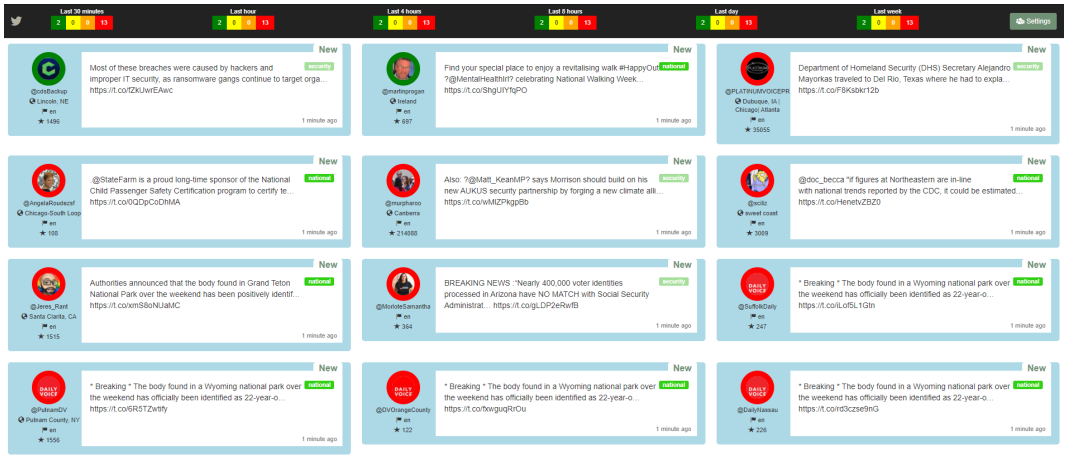


Figure 4.7: Web Page — second version — ordered by date.

Listing 7 Monitor Code - getMonitor

```
141         function getMonitor(tweetlist, date){
142
143
144             var monitorByColor = {
145                 green: tweetlist.filter(x => x.userRelevance >0.7 &&
                    ↪  x.userRelevance < 1 && x.dateCreated>=date).length,
146                 yellow: tweetlist.filter(x => x.userRelevance >0.5 &&
                    ↪  x.userRelevance < 0.7 && x.dateCreated>=date).length,
147                 orange: tweetlist.filter(x => x.userRelevance >0.3 &&
                    ↪  x.userRelevance < 0.5 && x.dateCreated>=date).length,
148                 red: tweetlist.filter(x => x.userRelevance >0 &&
                    ↪  x.userRelevance < 0.3 && x.dateCreated>=date).length
149             };
150             var html = "<div
                ↪  class='green'>"+monitorByColor.green+"</div>"+"<div
                ↪  class='yellow'>"+monitorByColor.yellow+"</div>"+"<div
                ↪  class='orange'>"+monitorByColor.orange+"</div>"+"<div
                ↪  class='red'>"+monitorByColor.red+"</div>";
151             return html;
152          }
```

4.5 Platform Evaluation

The evaluation of the platform consists in the comparison of the two versions, the first as a simpler version where the formulas were applied, and the second one as a most complete version based on the collected user feedback.
Two forms were sent to users in general (some of them usual *Twitter* users), and the card sort technique was applied to the collected feedback, as represented in figure 4.8.

4.5.1 Comparison of Three versions

The implemented features are shown in table 4.1, where a list of features is listed, even if some of them were not applied.
It is visible that the last version is the most complete, but some important features pointed by the users were not implemented, being part of the future work list.

Feature	First Version	Second Version
1 - Search Tweets	Yes	Yes
2 - Configure subjects to follow	No	Yes
3 - Color distinction by relevance	No	Yes
4 - Add new subjects	No	Yes
5 - Have temporal monitors	No	Yes
6 - Choose sort by date or relevance	No	Yes
7 - Filter by time	No	Yes
8 - Dark mode	No	Yes
9 - Adapt timezone	No	No
10 - Link to the Twitter post	No	No
11 - Language selection	No	No
12 - Show tweet media	No	No
13 - Change configuration when monitor is clicked	No	No

Table 4.1: Developed Features.

4.5.2 Card Sort

Card Sort technique, as described in 3.5.1, allows the validation of what is relevant for users, what should be more visible, or quick to access. In this case this technique was applied to collect important features in order to be implemented, to determine which are more important, and more users identify. This technique was applied according to forms answers, and interviews with some users of the first and second version of the platform, getting a list of features about what users refer to be important to have or an existing feature that they consider a problem, and then group the similar ideas in the same group of features that can be resolved by the same implementation.

By grouping the presented tester ideas, they were match to features presented in the feature list table 4.1, having the same numeration, as the aforementioned table:

- 2 — Configure subjects to follow;
- 3 — Color distinction by relevance;
- 4 — Add new subjects;
- 8 — Dark mode;
- 9 — Adapt timezone;
- 10 — Link to the Twitter post;
- 11 — Language selection;
- 12 — Show tweet media;
- 13 — Change time configuration according to monitor clicked.

Other features were identified not by the direct analysis of the presented features, but were considered as important to the user experience:

- 5 — Have temporal monitors;

- 6 — Choose sort by date or relevance;
- 7 — Filter by time;

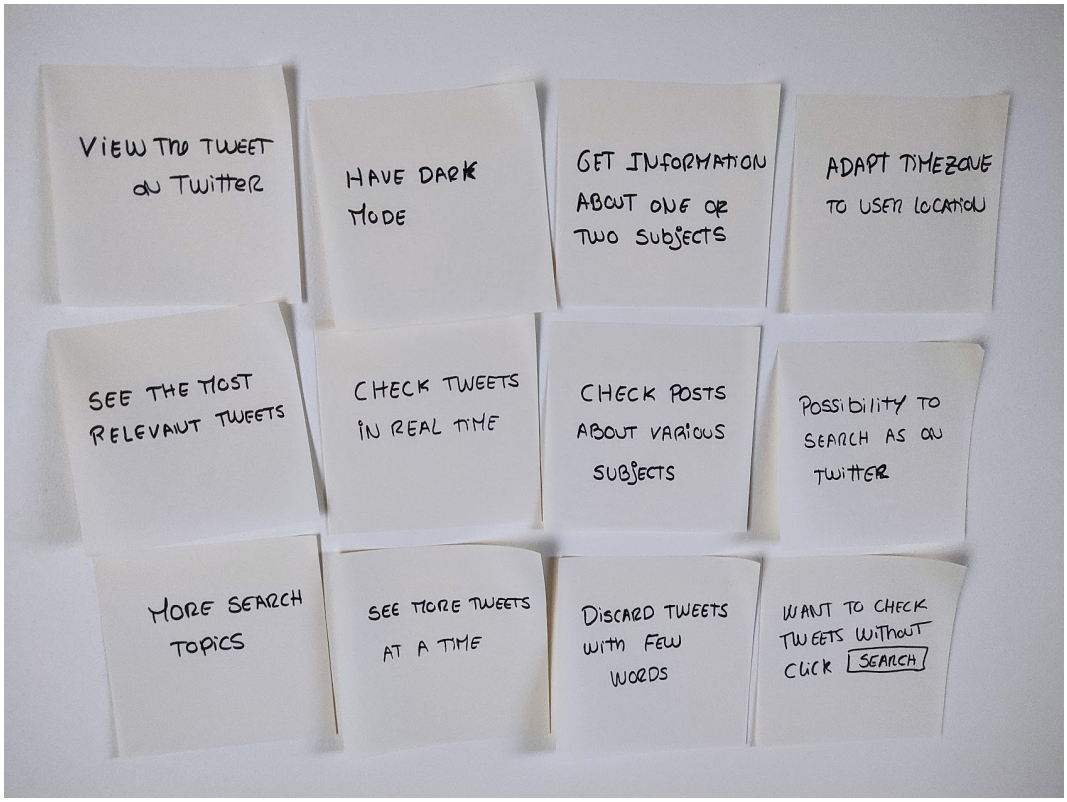


Figure 4.8: Card Sort features representation.

4.5.3 Forms

Two forms were filled by a group of users, some of them *Twitter* users, and others not. These users visited the page and had to do a search by some subject, giving their feedback about the interaction with the platform.

The first form has been done relative to the first version of the platform, developed in the context of this dissertation, and the second about the second version.

The first version, that can be consulted in appendix D had five answers, while the second, that can be consulted in appendix E had 28 answers.

In the first form all testers were Twitter users, while in the second not all testers were Twitter users, as can be visualized in figure 4.9, that almost half of the testers that participated were not familiar with Twitter, 40,07 per cent. But some of them visited Twitter, even not being users.

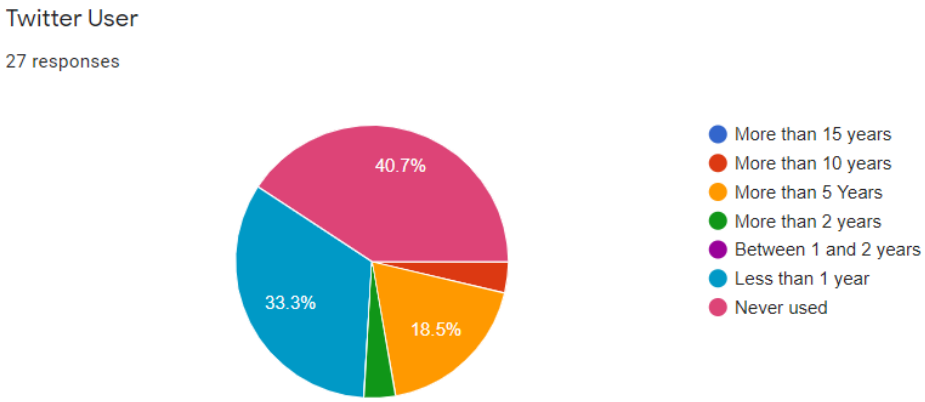


Figure 4.9: Twitter user form result graphical representation.

The **first version**, where User Experience, Layout, Responsivity, Post, posts Relevance, Post Information and Performance were evaluated by simple questions about how much the tester feels about the topic, being very satisfied, or nothing satisfied, showed that users were more satisfied with User Experience, Responsivity, Post and Performance. The worst results were in the Layout, the Post relevance and the Post information sections.

For the first version of the form, the measured parameters were evaluated from one to seven, where one is not satisfied and seven, very satisfied, as shown in figure 4.10.

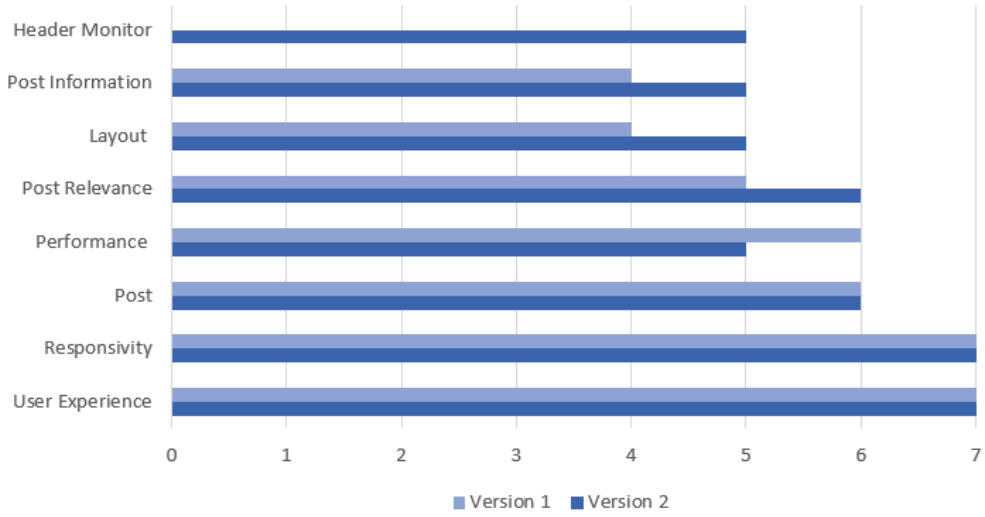


Figure 4.10: Users Average Evaluation graph in first and second version.

Through the analysis of the answers, it was noticed that some fields shown in the first version were not relevant to users.

About the relevance, and besides the page showing the most relevant posts first, many posts were presented, for example if the classification of one post is very high, and new posts occur in this subject, but these were not so relevant, the order can be a problem, as it is always showing the same posts in first place, keeping the user attention on these posts, so the idea of getting only the posts in a configurable number of minutes occurred, so when the post is older than the number of minutes configured, it will be hidden from the list.

About the layout, it was noticed that posts were taking up too much space, showing few posts at browser size. It was also found that it was difficult for the user to understand the result of the post classification, so the color scheme was placed, from green to yellow, orange and red, where green shows the most relevant and red the least relevant.

About the **second version**, the measure parameters were the same, the *responsivity* and user experience continued having a good evaluation. In terms of layout opinions were divided, some users compared the colors and appearance with twitter, arguing that Twitter is more fluid, and its layout more intuitive.

About post information besides having less unimportant information, some users revealed that will be important to open the tweet at Twitter and that the media of the post should be presented, by not having just the text, but the images and videos too.

4.6 Conclusions

Along the development process, important features were presented by users in the forms, that made the last version richer, and more attractive to the users. Some challenges with stop words, retweets, and irrelevant text, as posts only composed of URLs and hashtags, were removed, and increased the quality of the presented posts to the users, and performance of the page, once these are discarded before the insertion in the database.

Users were questioned specifically about the new monitors displayed in the top of the page, 40 per cent of the users liked the feature, but in the free text questions it was noticed that this functionality did not bring much help to users, only visual help, but several of them had the tendency to click the monitors where nothing occurred, and some even suggested as an improvement, to add the functionality of clicking on the monitors and then showing the posts from that timeline, some users had difficulty to understand this feature.

In terms of performance, the results were distinct, some users were expecting to see the posts about a new inserted subject in real time, and even the normal visualization of posts that already exists the time it leaves to process the tweets until they were shown in the screen was not very satisfactory for some users.
Sometimes the subjects were not having posts, because no new tweets about a given subject were being posted.

About the layout, some users did not understand the meaning of the colors assigned to the post relevance (green, yellow, orange, and red).

For the second version, as for the first one, the measured parameters were evaluated from one to seven, where one is nothing satisfied and seven, very satisfied, as shown in figure 4.10.
The last section of the second form, presented in appendix 5.4 was intended to compare the platform with twitter in order to understand if the answers in the first section are consistent with the second, and if some area needs to be improved.

Chapter 5

Conclusions and Future Work

5.1 Challenges

In the beginning it was aimed to continue the development of the proposed platform in PHP, in order to continue the work started before, but Node JS was an emerging runtime environment, so that the prototype was developed in Node JS, raising some challenges in terms of calls to database, and because some processes needed to run synchronously the synchronicity became a big challenge.
During the development of the platform some challenges occurred:

- The Twitter API was blocked by Twitter when tried to call new instances of the process, having many instances running at the same time, Twitter blocked the account, so that the keys needed to be renewed;
- Performance problems that had impact in the tests;
- By having many posts in the database the queries run slowly, so a deletion process needed to be implemented.

5.2 Strengths and Limitations

The new platform **strengths** are related with the improvements, and the features that were implemented based on users opinion:

- Possibility of having multiple subjects selected;
- New platform appearance;
- Availability of new configurations and filters;

The platform **limitations** are related with *scalability*, as the *webpage* does not support many accesses at the same time, and language, as only posts written in English are being gathered.

5.3 Future Work

Some features that might be added as future improvements were identified by users during the evaluation of the platform. Comments from users were collected at the end of the last evaluations form. The following features were tagged as important for the near future:

- Link to the Twitter post — Adding a link to Twitter Webpage for each of the posts displayed on the screen. It is essential to understand which field in JSON Twitter API structure represents the hyperlink, and then when the posts are collected, save that link in the database in a new field, that will be used to populate the HTML *hyperlink* `href` parameter;

- Language selection — Adding a new configuration, in the configuration menu, where we can configure the language of the displayed posts. Having in consideration that the posts will not be translated, it is necessary to prepare the stream to gather posts for the selected language, with a new table in the database being required in order to identify the possible languages to select and relate words to language. And at the stream process, filter posts by all the defined languages in the mentioned table (language);

- Show tweet media — It consists of showing images and videos available on tweets. The creation of a new table will be necessary in order to save all the media related to posts, and then show the saved media in the webpage in the correspondent component in HTML types, depending on if it is an image or a video;

- Change time configuration according to monitor clicked — It consists in permitting the users to click on the monitors, and see the posts being filtered by the number of hours that the monitor refers to. To implement this feature, the number of minutes filter (available on configurations menu) should be changed according to the monitor is selected. In terms of layout, it should be possible to distinguish what monitor is currently selected.

Beside the features determined by users some aspects needs to be improved, as:

- *Scalability*, so that many users can reach the platform. — The virtual machine resources will need to be improved, or a new virtual machine installed with the purpose of only running the web page process, so that the other two processes (streaming and evaluation) will run in a separated virtual machine, increasing the performance of accesses and preparing for a greater number of accesses;

- Some code refactoring is needed, so that it is easier to insert new features;

- Insert pagination in the end of the page — It consists in adding a navigation panel in the footer of the page where users can navigate when many posts are being displayed;

- Extract posts to Excel — It consists in having a button on the header, for example, or in the configurations panel, where the users can extract the displayed posts texts, and information to an Excel sheet;
- Create user area in order to save each user preferences, at this moment is in the session, so that in new sessions is always necessary to pick the preferences, in order to implement this feature a new table will be necessary to save the users. And in terms of front end the development of register and login pages will be needed;
- Have a *back-office* where the subjects can be configured, allowing to hide and delete subjects and configure other hard-coded parameters, such as the number of minutes a post is considered new, or default number of minutes the search is done. In terms of gathering posts, at this moment only the Twitter API is consulted, but could be interesting if other APIs were consulted, as Facebook, or LinkedIn, making the platform richer in terms of information variety.

5.4 Final Conclusions

Information is necessary to make decisions in many areas. Initially, this dissertation was focused in cyber-security studies, so that a prototype like the presented can be a useful tool to gather and filter the information that security engineers can base on to take decisions about new attacks and new techniques to prevent security issues. But many other areas can benefit from the usage of a system like this, such as journalism, education, medicine, and defense.

During the development of the two versions of the platform, user opinion was very important in order to find what would be important to include, and what is easy to use and what does not bring any value to users. The users that tested the platform were computer science people mainly, but other people were consulted too, from arts, history, companies management and finance areas, and from these people the most valuable issues were pointed, the main difficulties were presented as they are not so familiar with software testing or development as computer science people, as they manage to overcome some technical issues much more quickly, and do not give them as much importance.

The first developed version, as a simple version of what was proposed and presented, was important in order to test and improve the formulas and check the classification applied, then based on user feedback was possible to improve the platform having the presented final version, where more features were found to be implemented and marked as future work.

The database architecture used by the two developed versions was designed to receive posts from Twitter, but in the future other importation processes can be applied to populate the database with posts from other sources, making this platform richer in terms of

information.

Appendix

This chapter is composed of five appendixes that are important to complement the discussion:

- The first, where **Visual Design Principles** defined by Jon Yablonski [Jon20] are presented. There are twenty Laws of UX that represents the maxims and principles that must be taken into account when a web page is created;
- The second, is a content about user experience, about **Accessibility Guidelines** defined by Interaction Design Foundation [Fou20a] and points some important accessibility guidelines;
- The third, represents a ***Twitter JSON* Structure**;
- The fourth, represents the **First Version Evaluation Form** that was sent to the users to evaluate the first version of the platform;
- And the fifth represents the **Second Version Evaluation Form**, that was sent to the users to evaluate the second version of the platform.

A - Visual Design Principles

Law	Overview	Key Takeaways
Aesthetic Usability Effect	Users often perceive aesthetically pleasing design as design that's more usable.	Can lead to minor usability issues tolerance, and to lead people to believe that the aesthetics make things work better. In other way can mask usability problems, and prevent issues from being discovered during usability test.
Doherty Threshold	Productivity soars when a computer and its users interact at a pace (<400ms) that ensures that neither has to wait on the other.	Consists in keep user's attention and increases productivity, and using the perceived performance to increase response time and reduce the perception of waiting.
Fitts's Law	The time to acquire a target is a function of the distance to and size of the target.	The targets should have the enough size,the enough spacement between them, and being in and accessible area of the interface in order to be easily selected by user.
Hick's Law	The time it takes to make a decision increases with the number and complexity of choices.	Breakdown complex task in simple and smaller steps, highlight the most important options, and user progressive onboarding to minimise cognitive load for new users.
Jackob's Law	Users spend most of their time on other sites. This means that users prefer your site to work the same way as all the other sites they already know.	Users transfer expectations about a familiar product to another that appears similar. Based on other products that users already know (mental models) we can create experiences that focus users on tasks, instead of distracting them with new patterns. And when a new pattern is created we can provide the familiar pattern to the user until he is prepared to use the new one.
Law of Common Region	Elements tend to be perceived into groups if they are sharing an area with a clearly defined boundary.	Adding a border, or a background colour to an element to separate the elements by creating a boundary.
Law of Prägnanz	People will perceive and interpret ambiguous or complex images as the simplest form possible, because it is the interpretation that requires the least cognitive effort of us.	In complex shapes, or images the human eye finds the simplicity and the order.
Law of Proximity	Objects that are near, or proximate to each other, tend to be grouped together.	Proximity allows to group information with nearby objects. Leading to a better understanding of the presented information.

Law	Overview	Key Takeaways
Law of Similarity	The human eye tends to perceive similar elements in a design as a complete picture, shape, or group, even if those elements are separated.	Ensure that links and navigation systems are visually differentiated from normal text elements, and are consistently styled.
Law of Uniform Connectedness	Elements that are visually connected are perceived as more related than elements with no connection.	Group functions of a similar nature so they are visually connected via colors, lines, frames, or other shapes.
Miller's Law	The average person can only keep 7 (plus or minus 2) items in their working memory.	Chunking is an effective method of presenting groups of content in a manageable way. Organize content in groups of 5-9 items at a time.
Occam's Razor	Among competing hypotheses that predict equally well, the one with the fewest assumptions should be selected.	Analyze each element and remove as many as possible, without compromising the overall function.
Pareto Principle	The Pareto principle states that, for many events, roughly 80%-symbol of the effects come from 20%-symbol of the causes.	Focus the majority of effort on the areas that will bring the largest benefits to the most users.
Parkinson's Law	Any task will inflate until all of the available time is spent.	
Peak-End Rule	People judge an experience largely based on how they felt at its peak and at its end, rather than the total sum or average of every moment of the experience.	Consists in recording the most intense and "end" moments of users journey, in order to convert this moments most valuable, helpful or entertaining in the new product. Is important to note that people remember negative experiences more than positive ones.
Postel's Law	Be liberal in what you accept, and conservative in what you send.	Be empathetic, flexible, and tolerant to any number of actions the user could possibly take. This means accepting variable input from users, translating input to meet the requirements, defining boundaries for input, and providing clear feedback to the user.
Serial Position Effect	Users have a propensity to best remember the first and last items in a series.	Defends that the important information should be localised on the left or right part of the page, and the less important part in the middle.

Law	Overview	Key Takeaways
Tesler's Law	Tesler's Law, also known as The Law of Conservation of Complexity, states that for any system there is a certain amount of complexity which cannot be reduced.	
Von Restorff Effect	The Von Restorff effect, also known as The Isolation Effect, predicts that when multiple similar objects are present, the one that differs from the rest is most likely to be remembered.	Make important information or key actions visually distinctive.
Zeigarnik Effect	People remember uncompleted or interrupted tasks better than completed tasks.	Use progress bars for complex tasks to visually indicate when a task is incomplete, and thus increase the likelihood it will be completed.

B - Accessibility Guidelines

In the web page context, Interaction Design Foundation [Fou20a] points some important guidelines to take in account when a web page is developed:

- Ensure themes were designed for accessibility.
- Use alt text on content-enhancing images.
- Have a link strategy (i.e., describe the link before inserting it).
- Offer visual cues (e.g., PDF icons), underline links and highlight menu links on mouse-over.
- Improve visibility with careful colour selection and high contrast.
- Label fields and give descriptions to screen readers via tags.
- Make tab order visually ordered and assign a required or not required role to each field. Avoid the asterisk convention.
- Avoid tables for layout (only use for data presentation).
- Use proper HTML elements in lists (don't put them on the same line as text).
- Try using your design without a mouse. Scrolling can present difficulties.
- Validate markup using the W3 standards site [W3C20] to ensure all browsers can read your code.
- Avoid Flash.
- Offer transcriptions for audio resources, captions/subtitles for video.
- Make content readable – simpler language reaches more users.

C - Twitter JSON structure

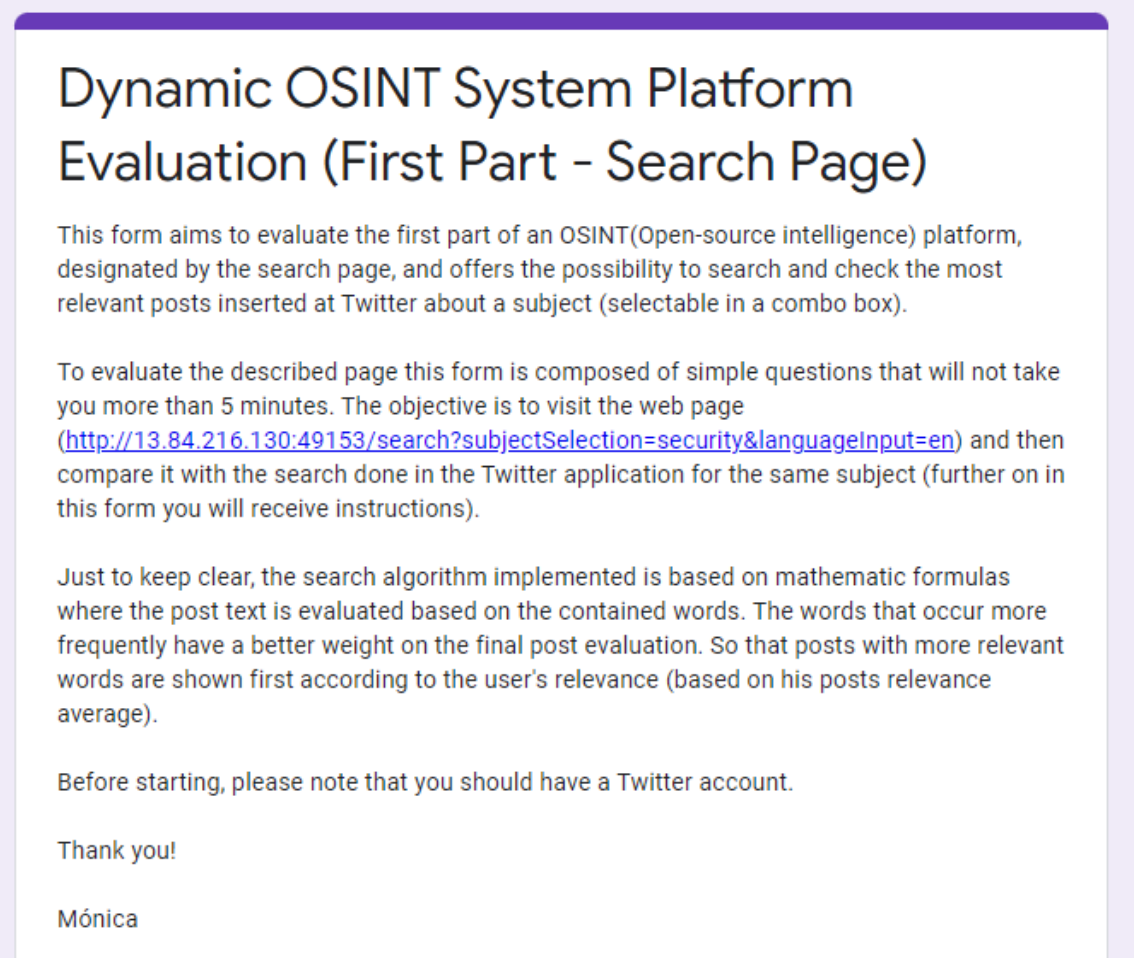
Listing 8 Tweet JSON Example

```
{  created_at: 'Mon Jun 28 15:17:27 +0000 2021',
  id: 1409531396075384800,
  id_str: '1409531396075384836',
  text: 'The tweet text comes here…',
  source: '<a href="https://mobile.twitter.com" rel="nofollow">Twitter Web
  ↪  App</a>',
  truncated: false,
    ...
  user:
   { id: 244078907,
     id_str: '244078907',
     name: 'Brutus',
     screen_name: 'screnname123',
     location: null,
     url: null,
     description: '11-year Navy veteran, proudly served 6 years on USS
     ↪  Enterprise. #VetsResistSquadron #VetsForBLM #NotNCAAProperty',
     translator_type: 'none',
     protected: false,
     verified: false,
     followers_count: 1684,
     friends_count: 3138,
     listed_count: 24,
     favourites_count: 65131,
     statuses_count: 20256,
     created_at: 'Fri Jan 28 13:28:14 +0000 2011',
     profile_background_color: 'C0DEED',
     profile_background_image_url:
     ↪  'http://abs.twimg.com/images/themes/theme1/bg.png',
     profile_background_image_url_https:
     ↪  'https://abs.twimg.com/images/themes/theme1/bg.png',
     profile_background_tile: false,
     profile_link_color: '1DA1F2',
     profile_sidebar_border_color: 'C0DEED',
     profile_sidebar_fill_color: 'DDEEF6',
     profile_text_color: '333333',
     profile_use_background_image: true,
     profile_image_url:
     ↪  'http://pbs.twimg.com/profile_images/1400471469977595912/gaMTb4UJ_normal.jpg',
     profile_image_url_https:
     ↪  'https://pbs.twimg.com/profile_images/1400471469977595912/gaMTb4UJ_normal.jpg'
     profile_banner_url:
     ↪  'https://pbs.twimg.com/profile_banners/244078907/1593796581'},
...
  quoted_status_id: 1409209984735449000,
  quoted_status_id_str: '1409209984735449090'

  ...
```

Listing 9 Tweet JSON Media

```
...
        extended_entities:
         { media:
            [ { id: 1409209680874852400,
                id_str: '1409209680874852352',
                indices: [ 223, 246 ],
                additional_media_info: { monetizable: false },
                media_url:
                 ↪  'http://pbs.twimg.com/ext_tw_video_thumb/.../pu/img/1.jpg',
                media_url_https:
                 ↪  'https://pbs.twimg.com/ext_tw_video_thumb/.../pu/img/1.jpg',
                url: 'https://t.co/hYC864ir8a',
                display_url: 'pic.twitter.com/hYC864ir8a',
                expanded_url:
                 ↪  'https://twitter.com/RightWingWatch/status/.../video/1',
                type: 'video',
                video_info:
                 { aspect_ratio: [ 16, 9 ],
                   duration_millis: 96067,
                   variants:
                    [ { content_type: 'application/x-mpegURL',
                        url: 'https://video.twim....1.m3u8?tag=12&mp4' },
                      { bitrate: 256000,
                        content_type: 'video/mp4',
                        url: 'https://video.twimg.com/.....1.mp4?tag=12'
                         ↪  },
                      { bitrate: 832000,
                        content_type: 'video/mp4',
                        url: 'https://video.twimg.com/.....1.mp4?tag=12'
                         ↪  },
                      { bitrate: 2176000,
                        content_type: 'video/mp4',
                        url: 'https://video.twimg.com/.....1.mp4?tag=12' }
                         ↪  ] },
                sizes:
                 { thumb: { w: 150, h: 150, resize: 'crop' },
                   medium: { w: 1200, h: 675, resize: 'fit' },
                   small: { w: 680, h: 383, resize: 'fit' },
                   large: { w: 1280, h: 720, resize: 'fit' } } } ] } },
     quote_count: 4313,
     reply_count: 6578,
     retweet_count: 1593,
     favorite_count: 4129,
...
```



Dynamic OSINT System Platform Evaluation (First Part - Search Page)

This form aims to evaluate the first part of an OSINT(Open-source intelligence) platform, designated by the search page, and offers the possibility to search and check the most relevant posts inserted at Twitter about a subject (selectable in a combo box).

To evaluate the described page this form is composed of simple questions that will not take you more than 5 minutes. The objective is to visit the web page (http://13.84.216.130:49153/search?subjectSelection=security&languageInput=en) and then compare it with the search done in the Twitter application for the same subject (further on in this form you will receive instructions).

Just to keep clear, the search algorithm implemented is based on mathematic formulas where the post text is evaluated based on the contained words. The words that occur more frequently have a better weight on the final post evaluation. So that posts with more relevant words are shown first according to the user's relevance (based on his posts relevance average).

Before starting, please note that you should have a Twitter account.

Thank you!

Mónica

D - First version Evaluation Form

Comprehension and agility

This section aims to evaluate the comprehension and agility of the search page.

Please access here: http://13.84.216.130:49153/search?subjectSelection=security&languageInput=en

Is composed of 8 required questions, that will measure the presented categories from 1 to 7. Where 1 is the worst rate, and 7 the best.

User Experience *

It is easy to complete the search and get results?

	1	2	3	4	5	6	7	
Very difficult	○	○	○	○	○	○	○	Very easy

Layout *

The layout is appellative?

	1	2	3	4	5	6	7	
Very unappealing	○	○	○	○	○	○	○	Very appealing

Responsivity *

Does the layout adapted correctly to your screen?

	1	2	3	4	5	6	7	
Showed very badly	○	○	○	○	○	○	○	Adapted correctly

Post *

Does the text of the posts match the selected subject?

	1	2	3	4	5	6	7	
Don't match	○	○	○	○	○	○	○	Clearly matches

Posts Relevance *

Did you consider the posts relevant to the selected subject?

	1	2	3	4	5	6	7	
Not very relevant	○	○	○	○	○	○	○	Very relevant

Post Information *

About the post representation. Do you consider you have all the information that you need? (User, Location, Language...)

	1	2	3	4	5	6	7	
Missing Information	○	○	○	○	○	○	○	Too much information

Performance *

Did your searches show results fast?

	1	2	3	4	5	6	7	
Very slow	○	○	○	○	○	○	○	Very fast

Recomendation *

How much you recommend this system to be used?

	1	2	3	4	5	6	7	
Not at all	○	○	○	○	○	○	○	Very much

Comparsion with Twitter

This section aims to evaluate the quality of the presented results by using this platform instead of a Twitter search.

At Platform:
(http://13.84.216.130:49153/search?subjectSelection=security&languageInput=en)
Please select some subject and click Search

At Twitter:
(https://twitter.com/)
Write the same word in search input, and then click enter.

User Experience *

Where is easier to complete the search and get results?

- ◯ This new platform
- ◯ Twitter
- ◯ Both

Layout *

Which has the most appealing layout?

- ◯ This new platform
- ◯ Twitter
- ◯ Both

Responsivity *

Which one best fits the screen?

- ◯ This new Platform
- ◯ Twitter
- ◯ Both

Post *

Which one returns more consistent results with the selected subject?

- ◯ This new Platform
- ◯ Twitter
- ◯ Both

Performance *

Whis one is faster to show results?

- ◯ This new Platform
- ◯ Twitter
- ◯ Both

Your Opinion (Optional)

This section is composed by some free and optional answers, where you can leave your comments and oppinion.

Thanks for helping!!

What would you add?

Your answer

What would you remove?

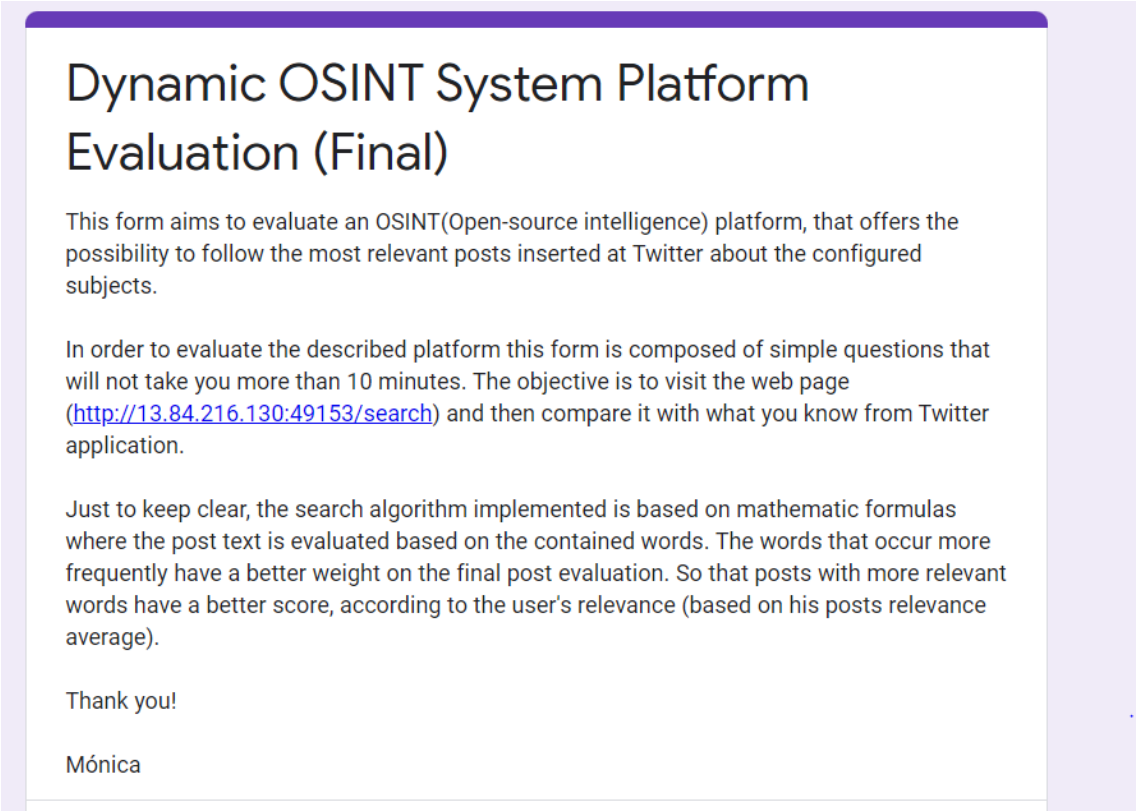
Your answer

Did you have difficulties or encountered problems? Can you descibe?

Your answer

Any other comments?

Your answer



Dynamic OSINT System Platform Evaluation (Final)

This form aims to evaluate an OSINT(Open-source intelligence) platform, that offers the possibility to follow the most relevant posts inserted at Twitter about the configured subjects.

In order to evaluate the described platform this form is composed of simple questions that will not take you more than 10 minutes. The objective is to visit the web page (http://13.84.216.130:49153/search) and then compare it with what you know from Twitter application.

Just to keep clear, the search algorithm implemented is based on mathematic formulas where the post text is evaluated based on the contained words. The words that occur more frequently have a better weight on the final post evaluation. So that posts with more relevant words have a better score, according to the user's relevance (based on his posts relevance average).

Thank you!

Mónica

E - Second version Evaluation Form

Dynamic OSINT System Platform Evaluation (Final)

monicarodriguesgil@gmail.com (not shared) Switch account

* Required

Your Identification

Before start, some data about you is important to catacterize your familiarity with Twitter, as your age, schooling and how many time you are using twitter.

Age *

Where your age fits?

- ◯ Between 18 and 25
- ◯ Between 26 and 35
- ◯ Between 36 and 45
- ◯ Between 46 and 55
- ◯ Between 56 and 65
- ◯ Between 66 and 75
- ◯ More than 75

Scooling *

What is your scolar degree?

- ◯ Graduated
- ◯ Post grade
- ◯ Master dregee
- ◯ Doctoral degree
- ◯ None of the above

Twitter User *

How many time you started using Twitter?

- ◯ More than 15 years
- ◯ More than 10 years
- ◯ More than 5 Years
- ◯ More than 2 years
- ◯ Between 1 and 2 years
- ◯ Less than 1 year
- ◯ Never used

Access Frequency *

How often did you use Twitter?

- ◯ More than 2 times a day
- ◯ Daily
- ◯ Weekly
- ◯ Sometimes I use it, but not frequenty
- ◯ Never Used

Comprehension and agility

This section aims to evaluate the comprehension and agility of the search page.

Please access here: http://13.84.216.130:49153/search

Is composed of 10 required questions, that will measure the presented categories from 1 to 7. Where 1 is the worst rate, and 7 the best.

User Experience

It is easy to change from dark mode to clear mode?

	1	2	3	4	5	6	7	
Very difficult	○	○	○	○	○	○	○	Very Easy

Layout *

The layout is appellative?

	1	2	3	4	5	6	7	
Very unappealing	○	○	○	○	○	○	○	Very appealing

User Experience *

It is easy to select the subjects you want to keep following?

	1	2	3	4	5	6	7	
Very difficult	○	○	○	○	○	○	○	Very easy

Responsivity *

Does the layout adapted correctly to your screen?

	1	2	3	4	5	6	7	
Showed very badly	○	○	○	○	○	○	○	Adapted correctly

Post *

Does the text of the posts match the selected subjects?

	1	2	3	4	5	6	7	
Don't match	○	○	○	○	○	○	○	Clearly matches

Posts Relevance *

Did you consider the posts relevant to the selected subjects?

	1	2	3	4	5	6	7	
Not very relevant	○	○	○	○	○	○	○	Very relevant

Post Information *

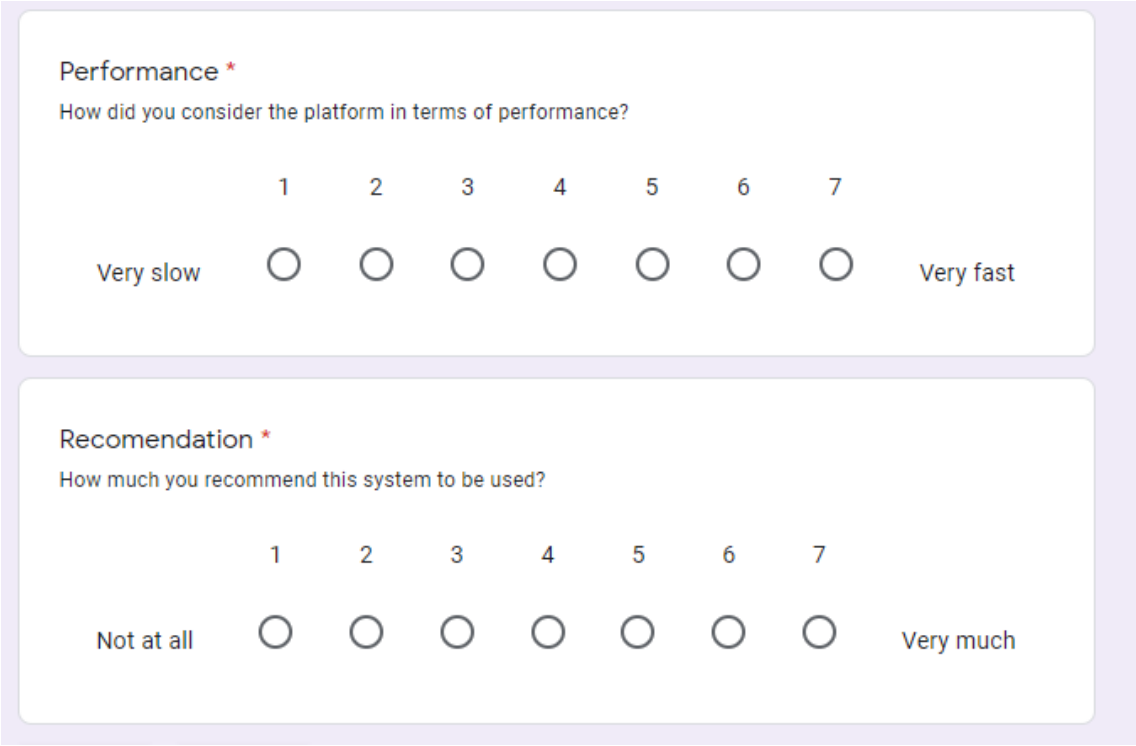
About the post representation. Do you consider you have all the information that you need? (User, Location, Language...)

	1	2	3	4	5	6	7	
Missing Information	○	○	○	○	○	○	○	Too much information

Header monitor *

Did you consider the top header monitors show relevant information about the relevance of the posts that are inserted by period of time?

	1	2	3	4	5	6	7	
Not very relevant	○	○	○	○	○	○	○	Very relevant


Performance *
How did you consider the platform in terms of performance?
1 2 3 4 5 6 7
Very slow
Very fast
Recomendation *
How much you recommend this system to be used?
1 2 3 4 5 6 7
Not at all
Very much

Comparsion with Twitter

This section aims to evaluate the quality of the presented results by using this platform instead of Twitter in order to follow one or more subjects of your choice.

At Platform:
(http://13.84.216.130:49153/search)
In configurations (available on top right menu) select some subjects and then reload the page.

At Twitter:
(https://twitter.com/)
Write the same word in search input, and then click enter.

User Experience

Where is easier to keep following results about a given subject?

- ◯ This new platform
- ◯ Twitter
- ◯ Both

Layout

Which has the most appealing layout?

- ◯ This new platform
- ◯ Twitter
- ◯ Both

Responsivity

Which one best fits the screen?

- ◯ This new Platform
- ◯ Twitter
- ◯ Both

Post

Which one returns more consistent results with the selected/searched subject(s)?

- ◯ This new Platform
- ◯ Twitter
- ◯ Both

Performance

Wish one is faster to show results?

- ◯ This new Platform
- ◯ Twitter
- ◯ Both

Your Opinion (Optional)

This section is composed by some free and optional answers, where you can leave your comments and oppinion.

Thanks for helping!!

What would you add?

Your answer

What would you remove?

Your answer

Did you have difficulties or encountered problems? Can you descibe?

Your answer

Any other comments?

Your answer

References

[AD03] Gregory D. Abowd Russell Beale Alan Dix, Janet E. Finlay (Author. *Human-Computer Interaction*. ISBN-13: 978-0130461094. Pearson, 3rd edition (September 30, 2003), 2003. Available from: https://www.amazon.com/Human-Computer-Interaction-3rd-Alan-Dix/dp/0130461091/ref=sr_1_2?dchild=1&qid=1631456656&refinements=p_27.27

[Adm21] PHP My Admin. Phpmyadmin [online]. 2021. Available from: https://www.phpmyadmin.net/ [cited 20/09/2021]. 49

[AT18] Taiar Redha (Editors) Ahram Tareq, Karwowski Waldemar. *Human Systems Engineering and Design*. ISBN-13: 978-3030020521. Springer; 1st ed. 2019 edition (October 16, 2018), 2018. Available from: https://www.amazon.com/Human-Systems-Engineering-Design-International-ebook/dp/B07JGR5M62. 26

[Ben10] David Benyon. *Designing Interactive Systems: A Comprehensive Guide to HCI and Interaction Design*. ISBN-13: 978-0321435330. Pearson Education Canada; 2 edition (March 16, 2010), 2010. Available from: https://www.amazon.com/Designing-Interactive-Systems-Comprehensive-Interaction/dp/0321435338. 25

[Ber14] Hal Berghel. Robert david steele on osint. *IEEE*, (14):76–81, 2014. 10

[Bla08] Simon Blackburn. *The Oxford Dictionary of Philosophy*. ISBN-13: 978-0199541430. Oxford University Press, 2008. Available from: https://www.oxfordreference.com/view/10.1093/acref/9780199541430.001.0001/acref-9780199541430. 30

[Blo20] Adobe Blog. Putting personas to work in ux design: What they are and why they're important [online]. 2020. Available from: https://theblog.adobe.com/putting-personas-to-work-in-ux-design-what-they\-are-and-why-theyre-important [cited 22/05/2020]. xvii, 33, 34

[bMTS17] Karin van Es by Mirko Tobias Schäfer. *The Datafied Society: Studying Culture through Data*. ISBN-13: 978-9462981362. Amesterdam University Press, 2017. Available from: https://www.amazon.com/Datafied-Society-Studying-Culture-through/dp/9462981361/ref=mt_hardcover?_encoding=UTF8&me=. 14

[Boo21] Boostrap. Bootstrap v3 [online]. 2021. Available from: https://getbootstrap.com/ [cited 21/09/2021]. 50

[Cao16] Jerry Cao. Paper prototyping: The 10-minute practical guide [online]. 2016. Available from: https://www.uxpin.com/studio/blog/paper-prototyping-the-practical-beginners-guide/ [cited 16/08/2020]. 38

[Chr17] Christian Martorella. Metagoofil package description [online]. 2017. Available from: https://tools.kali.org/information-gathering/metagoofil [cited 12/11/2017]. 11

[CO17] Thomas Owens Cyril Onwubiko. *Situational Awareness in Computer Network Defense: Principles, Methods and Applications 1st Edition*. ISBN-13: 978-1466601048. IGI Global, 2017. Available from: https://www.amazon.com/Situational-Awareness-Computer-Network-Defense/dp/1466601043/ref=sr_1_1?ie=UTF8&qid=1510616166&sr=8-1&keywords=Situational+Awareness+in+Computer+Network+Defense3A+Principles2C+Methods+and. 8

[Cot04] Viv Cothey. Web-crawling reliability. *JOURNAL OF THE AMERICAN SOCIETY FOR INFORMATION SCIENCE AND TECHNOLOGY*, (55(14)):1228–1238, 2004. 19

[(Ed07] Loch K. Johnson (Editor). *Handbook of Intelligence Studies (1st Edition)*. ISBN-13: 978-0415777834. Routledge, 2007. Available from: https://www.amazon.com/Handbook-Intelligence-Studies-Loch-Johnson/dp/0415777836. 5, 15

[EGH99] Michal Laclavik Marek Ciglan Emil Gatial, Zoltan Balogh and Ladislav Hluchy. Focused web crawling mechanism based on page relevance. *Institute of Informatics, Slovak Academy of Sciences*, 1999. 21

[Ele17] Eleven Paths. Foca [online]. 2017. Available from: https://www.elevenpaths.com/labstools/foca/index.html [cited 12/11/2017]. 11

[Exp17] Exploit Data Base. Exploit data base [online]. 2017. Available from: https://www.exploit-db.com/ [cited 12/11/2017]. 12

[Far17] Susan Farrell. From research goals to usability-testing scenarios: A 7-step method [online]. 2017. Available from: https://www.nngroup.com/articles/ux-research-goals-to-scenarios/ [cited 25/05/2020]. xvii, 35

[Fer16] Ivan Fernandes. Dynamic osint system sourcing from social networks. *UBI*, 2016. 1, 2, 17, 41, 42, 45, 48

[Fou18] Iteraction Design Foundation. How to recruit users for usability studies [online]. 2018. Available from: https://www.interaction-design.org/literature/article/how-to-recruit-users-for-usability-studies [cited 18/08/2020]. 39

[Fou20a] Interaction Design Foundation. Accessibility [online]. 2020. Available from: https://www.interaction-design.org/literature/topics/accessibility [cited 22/05/2020]. 25, 69, 73

[Fou20b] Interaction Design Foundation. Design thinking [online]. 2020. Available from: https://www.interaction-design.org/literature/topics/design-thinking [cited 19/05/2020]. 24

[Fou20c] Interaction Design Foundation. Putting some emotion into your design – plutchik's wheel of emotions [online]. 2020. Available from: https://www.interaction-design.org/literature/article/putting-some-emotion-into-your-design-plutchik-s-wheel-of-emotions [cited 20/05/2020]. 28

[Fou20d] Interaction Design Foundation. Responsive design [online]. 2020. Available from: https://www.interaction-design.org/literature/topics/responsive-design [cited 21/05/2020]. 26

[Fou21] Open JS Foundation. Node js [online]. 2021. Available from: https://nodejs.org/en/about/ [cited 20/09/2021]. 49

[Goo17] Google Hacking Database. Google hacking database (ghdb) [online]. 2017. Available from: https://www.exploit-db.com/google-hacking-database/ [cited 12/11/2017]. 12

[Goo20] Google. Google forms [online]. 2020. Available from: https://www.google.com/forms/ [cited 22/05/2020]. 33

[Gor20] Kelley Gordon. 5 principles of visual design in ux [online]. 2020. Available from: https://www.nngroup.com/articles/principles-visual-design/ [cited 18/08/2021]. 31

[Hub20] Usability Hub. Preference tests [online]. 2020. Available from: https://usabilityhub.com/product/preference-tests [cited 18/08/2020]. 39

[iai20] iainstitute. What is information architecture? [online]. 2020. Available from: https://www.iainstitute.org/what-is-ia [cited 15/08/2020]. 37

[IO20] Inga Wiele Isabell Osann, Lena Mayer. *The Design Thinking Quick Start Guide*. ISBN-13: 978-1119679899. Wiley; 1 edition (February 11, 2020), 2020. Available from: https://www.amazon.com/Design-Thinking-Quick-Start-Guide/dp/1119679893. 24

[JH12] Jian Pei Jiawei Han, Micheline Kamber. *Data Mining: Concepts and Techniques, Third Edition (The Morgan Kaufmann Series in Data Management Systems) 3rd Edition*. ISBN-13: 978-9380931913. MK, 2012. Available from: https://www.amazon.com/Data-Mining-Concepts-Techniques-Management/dp/0123814790/ref=sr_1_1?ie=UTF8&qid=1512343784&sr=8-1&keywords=The+Data+Mining3A+Concepts+and+Techniques. 18

[Jon20] Jon Yablonski. Laws of ux [online]. 2020. Available from: https://lawsofux.com/ [cited 14/01/2020]. 32, 69

[Jus17] Justin Nordine. osintframework.com [online]. 2017. Available from: http://osintframework.com/ [cited 12/11/2017]. 10

[KAL17] KALI. Kali [online]. 2017. Available from: https://www.kali.org/ [cited 12/11/2017]. 11

[Liu11] Bing Liu. *Web Data Mining - Exploring Hyperlinks, Contents, and Usage Data Second Edition*. ISBN-13: 978-3642194597. Springer, 2011. Available from: https://www.amazon.com/Web-Data-Mining-Data-Centric-Applications/dp/3642194591/ref=sr_1_1?ie=UTF8&qid=1512320482&sr=8-1&keywords=web+data+mining+liu. 19

[Lor16] Hoa Loranger. The negativity bias in user experience [online]. 2016. Available from: https://www.nngroup.com/articles/negativity-bias-ux/ [cited 21/05/2020]. 30, 31

[Low13] Travis Lowdermilk. *User-Centered Design: A Developer's Guide To Building User-Friendly Applications*. ISBN-13: 978-1449359805. O'Reilly Media; 1 edition (April 14, 2013), 2013. Available from: https://www.amazon.com/User-Centered-Design-Developers-User-Friendly-Applications/dp/1449359809. 23, 24, 33, 35, 38, 39

[luc20] lucidchart. The visual workspace for remote teams [online]. 2020. Available from: https://app.lucidchart.com/ [cited 12/08/2020]. xvii, 36

[Mah03] Manda Mahoney. Harvard research ideas: The subconscious mind of the consumer (and how to reach it) [online]. 2003. Available from: https://hbswk.hbs.edu/item/the-subconscious-mind-of-the-consumer-and-how-to-reach-it [cited 19/05/2020]. 29

[Mic21] Microsoft. Azure platform [online]. 2021. Available from: https://azure.microsoft.com/en-us/ [cited 20/09/2021]. 49

[Nic17a] Nick Babich. A beginner's guide to information architecture for ux designers [online]. 2017. Available from: https://blog.adobe.com/en/2017/11/20/a-beginners-guide-to-information-architecture-for-ux-designers.html#gs.d0jb5n [cited 16/08/2020]. 37

[Nic17b] Nick Babich. Prototyping 101: The difference between low-fidelity and high-fidelity prototypes and when to use each [online]. 2017. Available from: https://blog.adobe.com/en/2017/11/29/prototyping-difference-low-fidelity-high-fidelity-prototypes-use.html#gs.dos7gy [cited 16/08/2020]. 38

[Nic17c] Bogdan Tiganoaia; Alexandra Cernian; Andrei Niculescu. The use of social platforms and personal data protection – an exploratory study. *IEEE*, pages 1 – 5, 2017. 6

[Nie00] Jakob Nielsen. Why you only need to test with 5 users [online]. 2000. Available from: https://www.nngroup.com/articles/why-you-only-need-to-test-with-5-users/ [cited 18/08/2020]. xvii, 40

[Nie20a] Jakob Nielsen. How to conduct a heuristic evaluation [online]. 2020. Available from: https://www.nngroup.com/articles/how-to-conduct-a-heuristic-evaluation/ [cited 18/08/2020]. 39

[Nie20b] Jakob Nielsen. Putting a/b testing in its place [online]. 2020. Available from: https://www.nngroup.com/articles/putting-ab-testing-in-its-place/ [cited 1/08/2020]. 39

[Nik19] Anton Nikolov. Design principle: Consistency [online]. 2019. Available from: https://uxdesign.cc/design-principle-consistency-6b0cf7e7339f [cited 21/05/2020]. 23

[nma17] nmap. nmap [online]. 2017. Available from: https://nmap.org/ [cited 12/11/2017]. 11

[NN20] Don Norman and Jakob Nielsen. The definition of user experience (ux) [online]. 2020. Available from: https://www.interaction-design.org/literature/topics/ux-design [cited 19/08/2020]. 23

[Nor03] Don Norman. *Emotional Design: Why We Love (or Hate) Everyday Things*. 978-0465051359. Basic Books; 1 edition (December 24, 2003), 2003. Available from: https://www.amazon.com/dp/0465051359?tag=donnormanA. 29

[Nor13] Don Norman. *Design of Everyday Things*. ISBN-13: 978-0465050659. Basic Books; Revised edition (November 5, 2013), 2013. Available from: https://www.amazon.com/Design-Everyday-Things-Revised-Expanded/dp/0465050654. 24, 27, 30

[NPM21a] NPM. Node js - natural [online]. 2021. Available from: http://naturalnode.github.io/natural/ [cited 20/09/2021]. 51

[NPM21b] NPM. Node js - stopword [online]. 2021. Available from: https://www.npmjs.com/package/stopword [cited 20/09/2021]. 51

[NPM21c] NPM. Node js - twitter api [online]. 2021. Available from: https://www.npmjs.com/package/twitter [cited 20/09/2021]. 51

[Org11] World Heaslth Organization. World report on disability 2011. *WHO - World Health Organization*, 2011. 25

[Pat17] Paterva. Maltego framework [online]. 2017. Available from: https://www.paterva.com/ [cited 12/11/2017]. 10, 11

[Pen16] Alan Pennington. *The Customer Experience Book: How to design, measure and improve customer experience in your business*. ISBN-13: 978-1292148465. FT Press; 1 edition (September 20, 2016), 2016. Available from: https://www.amazon.com/Customer-Experience-Book-customer-experience/dp/1292148462. 35

[RZ14] Fabio Sangiacomo Rodolfo Zunino, Roberto Surlinelli. An analyst-adaptive approach to focused crawlers. *2013 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining*, pages 1073–1077, 2014. 17, 19, 20

[SC99] Byron Dom Soumen Chakrabarti, Martin van den Berg. Focused crawling: a new approach to topic-specific web resource discovery. *Computer Networks: The International Journal of Computer and Telecommunications Networking*, (31):1623–1640, 1999. 19

[Sch09] Eric Schaffer. Beyond usability: Designing web sites for persuasion, emotion, and trust [online]. 2009. Available from: https://www.uxmatters.com/mt/archives/2009/01/beyond-usability-designing-web-sites-for-persuasion-emotion-and-trust.php [cited 19/05/2020]. 29

[Sec17] Security Through Education. The social engineering framework [online]. 2017. Available from: https://www.social-engineer.org/framework/se-tools/computer-based/social-engineer-toolkit-set/ [cited 12/11/2017]. 13

[sho17] shodan. Shodan - monitor network security [online]. 2017. Available from: https://www.shodan.io [cited 12/11/2017]. 11

[SS13] Florian Schaurer and Jan Störger. The evolution of open source intelligence (osint). *Journal of U.S. Intelligence Studies*, pages 53–56, 2013. 9

[Sur20] SurveyMonkey. Surveymonkey [online]. 2020. Available from: https://www.surveymonkey.com/ [cited 22/05/2020]. 33

[Twi21a] Twitter. Consuming streaming data [online]. 2021. Available from: https://developer.twitter.com/en/docs/tutorials/consuming-streaming-data [cited 21/09/2021]. 55

[Twi21b] Twitter. Search tweets: Standard v1.1 [online]. 2021. Available from: https://developer.twitter.com/en/docs/twitter-api/v1/tweets/search/api-reference/get-search-tweets [cited 21/09/2021]. 55

[usa20] usability.gov. Card sorting [online]. 2020. Available from: https://www.usability.gov/how-to-and-tools/methods/card-sorting.html [cited 16/08/2020]. xvii, 37

[W3C20] W3C. W3c standards [online]. 2020. Available from: https://www.w3.org/standards/ [cited 22/05/2020]. 73

[Whi17] Kathryn Whitenton. Tree testing: Fast, iterative evaluation of menu labels and categories [online]. 2017. Available from: https://www.nngroup.com/articles/tree-testing/ [cited 16/08/2020]. 38

[XAM21] XAMPP. Apache [online]. 2021. Available from: https://www.apachefriends.org/index.html [cited 20/09/2021]. 49

[xte20] xtensio. User persona template and examples [online]. 2020. Available from: https://xtensio.com/user-persona/ [cited 22/05/2020]. 33

[Zal03] Gerald Zaltman. *How Customers Think*. ISBN-13: 978-1578518265. Harvard Business School Press; 1 edition (February 21, 2003), 2003. Available from: https://www.amazon.com/How-Customers-Think-Essential-Insights/dp/1578518261. 26