UNIVERSIDADE DO ESTADO DO RIO GRANDE DO NORTE – UERN

FACULDADE DE CIÊNCIAS EXATAS E NATURAIS – FANAT

DEPARTAMENTO DE INFORMÁTICA – DI

Thamires Araújo Magalhães de Lucena

# Um *Crawler* para Extração e a Análise de Dados do Fórum do Warframe

MOSSORÓ - RN

2018

Thamires Araújo Magalhães de Lucena

# Um *Crawler* para Extração e a Análise de Dados do Fórum do Warframe

Monografia apresentada à Universidade do Estado do Rio Grande do Norte como um dos pré-requisitos para obtenção do grau de bacharel em Ciência da Computação, sob orientação do Prof. Dr. Marcelino Pereira dos Santos Silva.

MOSSORÓ - RN

2018

A663c Araújo Magalhães de Lucena, Thamires
Um Crawler para Extração e a Análise de Dados do Fórum do Warframe. / Thamires Araújo Magalhães de Lucena. - Mossoró, 2018.
37p.

Orientador(a): Prof. Dr. Marcelino Pereira dos Santos Silva.
Monografia (Graduação em Ciência da Computação). Universidade do Estado do Rio Grande do Norte.

1. Análise de Dados. 2. Warframe. 3. Scrapy. 4. Weka. 5. Correlação. I. Pereira dos Santos Silva, Marcelino. II. Universidade do Estado do Rio Grande do Norte. III. Título.

Thamires Araújo Magalhães de Lucena

**Um *Crawler* para Extração e a Análise de Dados do Fórum do Waframe**

Monografia apresentada como pré-requisito para obtenção do título de Bacharel em Ciência da Computação da Universidade do Estado do Rio Grande do Norte – UERN, submetida à aprovação da banca examinadora composta pelos seguintes membros:

Aprovada em: 07/12/2018.

Banca Examinadora

Prof. Dr. MARCELINO PEREIRA DOS SANTOS SILVA
Universidade do Estado do Rio Grande do Norte - UERN

PROF. ALYSSON MENDES DE OLIVEIRA
Universidade do Estado do Rio Grande do Norte - UERN

Prof. Dr. CARLOS HEITOR PEREIRA LIBERALINO
Universidade do Estado do Rio Grande do Norte - UERN

*À minha mãe, Selda Marta, e meus professores.*

## AGRADECIMENTOS

Gostaria de agradecer primeiramente aos meus pais, por terem me dado a vida e a oportunidade de estar aqui. Mas especialmente a minha mãe, dona Selda Marta, que sempre fez tudo que podia para que eu pudesse ter melhores oportunidades para estudar, que sempre me garantiu tudo que precisei, por ter tido toda paciência comigo durante essa etapa da minha vida. Muito obrigada, minha mãe, amo muito a senhora!

Agradecer também a todos os meus professores, pois sempre que precisei, estavam dispostos a me ajudar, a esclarecer minhas dúvidas, a me guiar. Quero agradecer especialmente aos professores Marcelino e Alexsandra, por terem me aconselhado e me ajudado muito em uma época que passei por problemas psicológicos.

Aos amigos que fiz durante a faculdade, que estávamos juntos nos bons e maus momentos. Especialmente os momentos que passei no PET ao lado de Giovana, Wilton e Álvaro. Mas também quero agradecer a Ísis e Yakamuri, por me apoiarem e não me deixarem desistir. Vocês são incríveis!

Quero agradecer também a todos que me apoiaram quando eu decidi deixar de cursar Direito e vim atrás do meu sonho em Ciência da Computação, especialmente a minha mãe, meu irmão e meu amigo Mário Bandeira, que até lógica de programação me ensinou.

Agradecer também ao meu namorado, Aristóteles Linine, por todo amor e paciência que teve comigo durante esse período conturbado que foi o final da graduação. Lembre-se, *Here Until Forever*.

Agradecer a minha psicóloga, Maria Tereza, que me acompanhou nos meus momentos mais difíceis e que me ajudou a crescer e me desenvolver como ser humano. A senhora é um ser iluminado, o mundo deveria ter mais pessoas como você.

Quero agradecer aos meus amigos da Tribo Inapaiê. Vocês são uns loucos, mas sempre estiveram comigo, seja jogando ou só conversando pelo discord. Isso sem falar nos nossos encontros, as nossas InapaieCon, momentos que sempre irei guardar.

Agradecer a todas as bandas que me proporcionaram horas de músicas que me ajudaram bastante. Especialmente Gojira e Dream Theater, pois ouvindo suas músicas foi quando tive as melhores ideias e soluções para os problemas que estava passando.

A Digital Extremes por desenvolvido o jogo da minha vida, Warframe. Mesmo fazendo pouco mais de um ano que eu jogo, eu ainda me sinto encantada como me senti desde a primeira vez que joguei.

Quero agradecer a todos que me auxiliaram nas dificuldades que tive neste trabalho, em especial ao professor Alysson, Sedir, Thiago Jobson e Wanderson. Vocês me ajudaram a seguir pelo melhor caminho para resolver os problemas que tive durante o desenvolvimento do *crawler* e na parte da correlação estatística.

E por último, mas não menos importante, quero agradecer as minhas duas gatas, Rukia e Furiosa, especialmente a Rukia. Mesmo elas não entendendo nenhuma dessas palavras, elas sempre estiveram comigo, demonstrando um amor e carinho únicos.

“Sonhe não com o que você é, mas
com o que você quer ser.”

Lotus, *Warframe*

## RESUMO

A indústria de jogos está em constante crescimento e com isso aumenta também a quantidade de dados produzidos pelos jogadores. Tais dados podem ser a quantidade de jogadores simultâneos, o dinheiro que foi gasto em compras de moedas virtuais, interação dos jogadores em redes sociais, dentre vários outros. Com isso em mente, é possível utilizar o processo de Descoberta de Conhecimento em Banco de Dados para encontrar padrões válidos nestes dados. O presente trabalho busca fazer uma análise para saber se existe uma relação entre a atividade de jogadores no fórum do Warframe e a quantidade média de jogadores. Para conseguir extrair os dados do fórum de forma automática, foi desenvolvido um *web crawler* utilizando Scrapy, um framework Python para extrair dados estruturados de sites. A análise foi realizada por meio do cálculo de correlação e do software Weka.



**Palavras-chave:** Análise de Dados, Warframe, Scrapy, Weka, Correlação.

## ABSTRACT

The gaming industry is constantly growing, and this also increases the amount of data produced by players. Such data can be the number of simultaneous players, the money that was spent on virtual currencies, interaction of the players in social networks among several others. It is possible to use the Knowledge Discovery in Databases to find valid patterns in this data. The present work aims to make an analysis to know if there is a relation between the activity of the players in the forum of the Warframe and the average amount of players. In order to extract the data from the forum automatically, a web crawler was developed using Scrapy, a Python framework to extract structured data from websites. The analysis was performed by means of correlation calculations and Weka software.



**Keywords:** Data Analysis, Warframe, Scrapy, Weka, Correlation.

**SUMÁRIO**

## LISTA DE SIGLAS

| | |
|---|---|
| ARFF | *Attribute-Relation File Format* |
| CSS | *Cascading Style Sheets* |
| CSV | *Comma-Separated Values* |
| HTML | *Hypertext Markup Language* |
| KDD | *Knowledge Discovery in Databases* |
| RPG | *Role-playing Game* |
| RTS | *Real-Time Strategy* |
| TPS | *Third Person Shooter* |
| WEKA | *Waikato Environment for Knowledge Analysis* |

## LISTA DE FIGURAS

## LISTA DE TABELAS

## LISTA DE EQUAÇÕES

# 1. INTRODUÇÃO

A indústria de jogos está em constante crescimento, segundo o portal Statista (2018) a previsão para o faturamento em 2018 é de aproximadamente 115 bilhões de dólares, crescendo e podendo chegar a alcançar 138 bilhões em 2021. Este faturamento é baseado em duas grandes fontes: o hardware (consoles, telas, controles e outros acessórios) e o software, que são os jogos em si.

Com esta crescente demanda, cresce também a quantidade de dados, informações produzidas pelos jogadores. Tais dados podem incluir atividade, quantidade de jogadores simultâneos, gasto em dinheiro virtual dentro do jogo, interação dos jogadores em redes sociais, dentre outros.

Tendo isso em vista, é possível fazer uma análise nesses dados, utilizando o processo de Descoberta de Conhecimento em Banco de Dados (KDD, *Knowledge Discovery in Databases*), descrito por Fayyad et al. (1996) como o conjunto de técnicas e processos realizados nos dados para descobrir padrões válidos, novos, úteis e compreensíveis nos dados.

O objetivo principal deste trabalho é fazer uma análise entre a atividade do fórum no Warframe e a quantidade média de jogadores na plataforma PC, disponibilizado pelo Steam Charts, um site que mostra a estatística de todos os jogos da Steam. Também será feita uma outra análise com os dados coletados para verificar padrões de atividade no fórum. Para realizar este objetivo, foi proposto desenvolver um *web crawler* a fim de percorrer as páginas do fórum para extrair os dados, seguindo da análise dos dados coletados.

Este trabalho está disposto da seguinte maneira: No capítulo 2 é apresentada a fundamentação teórica, onde todos os principais conceitos e ferramentas utilizadas são apresentados. O capítulo 3 é a explicação de como foi feito o trabalho, como as tecnologias foram utilizadas, a metodologia. Os resultados são apresentados no capítulo 4. No capítulo 5 são apresentadas as conclusões e trabalhos futuros.

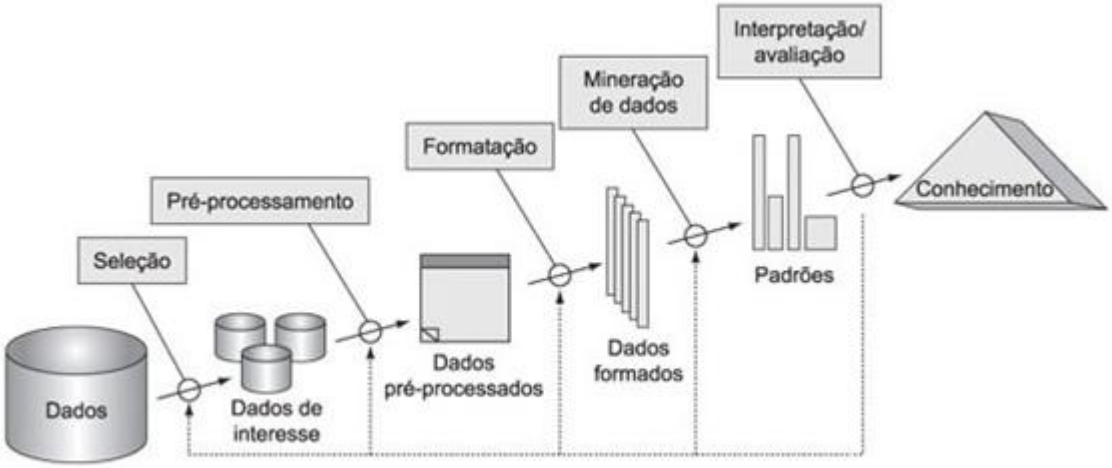
# 2. FUNDAMENTAÇÃO TEÓRICA

Esta seção é destinada a esclarecer os principais conceitos que serão abordados neste trabalho: Mineração de Dados; Python; Warframe. Também serão abordados os trabalhos relacionados e uma breve comparação entre com este trabalho.

## 2.1. Mineração de Dados

O processo de Descoberta de Conhecimento em Banco de Dados (do termo inglês *Knowledge Discovery in Databases*, KDD) na visão de Fayyad et al. (1996) é todo o conjunto de técnicas e processos realizados nos dados para descobrir padrões válidos, novos, úteis e compreensíveis nos dados. Tais processos são dinâmicos e interativos, podendo haver loops entre os processos, conforme visto na Figura 1.

Figura 1 - Processo KDD




Fonte: Fayyad et al. (1996)

Todo o processo KDD pode ser descrito como:

1. O primeiro passo é conhecer os dados que serão trabalhados e estabelecer o que se busca através do processo KDD;

2. Seleção dos dados: Criação de um conjunto de dados alvos a serem trabalhados, focar num conjunto ou subconjunto dos dados onde o processo de descoberta será aplicado;

3. Limpeza e pré-processamento dos dados: realizadas operações de remoção e tratamento de ruídos nos dados, escolha de estratégias para manipular campos ausentes, formatação dos dados para que se adequem a ferramenta de mineração que será utilizada;

4. Redução dos dados e formatação: encontrar características úteis para representar os dados de acordo com o objetivo da tarefa, para haver uma redução no número de variáveis ou das instâncias a serem utilizadas no conjunto de dados;

5. Mineração de dados: Selecionar o método que será utilizado para descobrir os padrões nos dados tratados, fazendo uma análise exploratória do modelo ajustando-o às necessidades dos dados a serem minerados;

6. Interpretação dos dados: visualização dos padrões extraídos, podendo voltar aos passos anteriores para mais interações;

7. Conhecimento descoberto: adicionar o conhecimento adquirido ao sistema, ou documentá-lo e informar as partes interessadas.

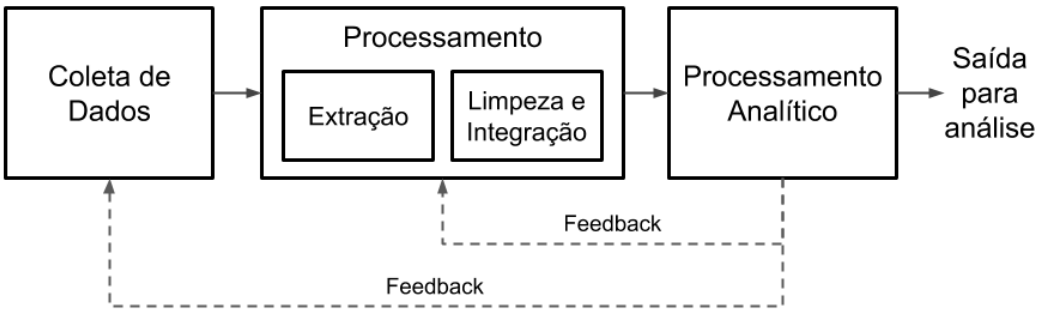
O processo de mineração de dados envolve uma variedade de outros processos e estudos. Para Aggarwal (2015), a mineração de dados é o estudo da coleta, limpeza, processamento, análise e obtenção de informações úteis, um termo amplo para descrever o processamento de dados, sendo composta por três fases, podendo ser observadas na Figura 2.

1. Coleta de dados: Pode exigir o uso de hardware especializado, como rede de sensores, mão-de-obra manual, ou a coleta de pesquisas de usuários ou ferramentas de software como um mecanismo de rastreamento Web para coletar documentos;

2. Extração e limpeza dos dados: Geralmente, os dados coletados não estão em um formato adequado aos algoritmos de mineração, nesta fase os dados são tratados para que se adequem ao formato utilizado;

3. Processamento analítico e algoritmos: Projetar ou escolher métodos analíticos eficazes a partir dos dados coletados.

Figura 2 - Processo de mineração de dados




Fonte: Aggarwal (2015)

Mineração de dados fornece uma maneira do computador aprender a tomar decisões com os dados. Essas decisões podem prever o tempo de amanhã, bloquear um e-mail spam na sua caixa de entrada, detectar a linguagem de um site, ou achar um novo romance em site de encontros. Mineração de dados é parte de algoritmos, estatísticas, engenharia, otimização e ciência da computação. Podendo ainda utilizar conceitos e conhecimentos de outras áreas como linguística, neurociência ou urbanismo (LAYTON, 2015). Pode ser vista como o resultado natural da evolução da tecnologia da informação. O desenvolvimento inicial dos mecanismos de coleta de dados e de criação de bancos de dados serviu como um pré-requisito para o desenvolvimento posterior de mecanismos eficazes de armazenamento e recuperação de dados, bem como processamento de consultas e transações (HAN et al., 2012).

Existem diversas ferramentas que auxiliam no processo de mineração de dados, a escolhida para se utilizar neste trabalho foi o Weka (*Waikato Environment for Knowledge Analysis*). Desenvolvido pela Universidade de Waikato na Nova Zelândia, o Weka (FRANK et al., 2016) é uma coleção de algoritmos de aprendizado de máquina e ferramentas de pré-processamento de dados, fornecendo suporte para todo o processo de mineração de dados experimentais, incluindo a preparação dos dados de entrada, a avaliação estatística dos esquemas de aprendizado e a visualização dos dados de entrada e o resultado do aprendizado.

O processo de mineração e análise de dados envolve diversos conhecimentos, além dos conhecimentos computacionais. Um dos mais importantes é o conhecimento estatístico. Para Freund (2006), estatística envolve tudo que trata com coleta, processamento, interpretação e apresentação de dados.

## 2.2. Python

Segundo a documentação (PYTHON, 2018) Python é uma linguagem de programação orientada a objetos, interpretada e interativa, com modos incorporados, dinamicamente tipada, alto nível, com tipos de dados e classes dinâmicas. Python é uma linguagem poderosa com uma sintaxe muito clara e limpa. Possui uma interface com muitas chamadas para sistemas e bibliotecas, bem como para vários sistemas Windows, e é extensiva em C/C++. É também usada como uma linguagem de extensão para aplicações que precisam de uma interface programável. Finalmente, Python é portável: funciona em várias versões do Unix, no Mac e no Windows.

A técnica utilizada para coleta de dados foi *Web Scraping*, que é um termo para se referir ao uso de um programa para fazer download e processar conteúdos Web (SWEIGART, 2015). Para isso foi utilizado o Scrapy, um framework para rastrear sites e extrair dados estruturados que podem ser usados em várias aplicações úteis, como mineração de dados, processamento de informação, entre outras. Apesar do Scrapy ter sido desenvolvido inicialmente para web *scraping*, ele também pode ser usado para extrair dados usando APIs (*Application Programming Interface*) ou com o propósito geral de *web crawler* (SCRAPY, 2018).

Para tratamento dos dados, o Python dispõe do pacote Pandas. Este pacote possui estrutura de dados rápidas, flexíveis e expressivas, projetadas para tornar o trabalho com dados mais fácil e intuitivo (PANDAS, 2018). Foi utilizado o Jupyter Notebook para se trabalhar com o Pandas, que é um ambiente de computação interativo com interface Web que permite o usuário criar e editar conteúdos de documentos notebook (uma representação de todo conteúdo visível no aplicativo da Web, incluindo entradas e saídas do código) (JUPYTER, 2018).

## 2.3. Warframe

Em *Level Up! The Guide to Great Video Game Design* (ROGERS, 2010) encontramos algumas definições importantes sobre jogos a serem exploradas antes de definirmos *Warframe*, tais como:

Tiro em Terceira Pessoa (do inglês *Third Person Shooter*, TPS): Um atirador onde a câmera fica localizada por trás do jogador, permitindo uma visão parcial ou completa dos outros jogadores ao seu redor. Apesar da visão mais ampla, a ênfase da jogabilidade ainda permanece no disparo.

Aventura: jogos de aventura tem foco em solução de enigmas, coleção de itens e gestão de inventário. Anteriormente, jogos de aventura eram apenas baseados em textos.

RPG, *Role-playing game* (termo utilizado sem tradução para o português e demais idiomas): É um subgênero do gênero aventura. É baseado nos jogos de interpretação que usam papel e caneta, como *Dungeons and Dragons*. Os jogadores escolhem uma classe e aumentam suas estatísticas de habilidades através do combate, exploração e tesouros. Os personagens podem ser específicos ou classes genéricas.

*Warframe* é um jogo de computador gratuito para jogar (do termo inglês *free-to-play*) desenvolvido e operado pela Digital Extremes Ltd, um jogo de ação TPS com elementos de RPG e temática de ficção científica espacial.

## 2.4. Trabalhos Relacionados

Pingen (2014) desenvolveu um agente automatizado que utiliza um modelo de previsão de estratégia que utilizou dados baseados em replays de partidas entre jogadores de *Starcraft: Brood War,* que é uma expansão do jogo *Starcraft* desenvolvido pela *Blizzard Entertainment*. É um jogo de estratégia em tempo real (*Real-Time Strategy*, RTS) onde os jogadores precisam coletar e administrar recursos e construções para fazer um exército que derrote o exército e as construções inimigas. A heurística utilizada para selecionar uma contra estratégia foi baseada na noção de que uma estratégia ofensiva é melhor contra uma defensiva e vice-versa. Os resultados obtidos mostraram que esta abordagem é promissora, mas ainda não é suficiente para competir contra os robôs desenvolvidos para *Starcraft: Brood War*.

Dringus e Ellis (2004) em *Using data mining as a strategy for assessing asynchronous discussion forums* utilizaram técnicas de mineração de dados em um fórum assíncrono para melhorar a capacidade de um instrutor avaliar uma discussão

encadeada. Partindo da problemática que os instrutores possuem em avaliar seus alunos em fóruns assíncronos, foram identificados os indicadores de participação que os instrutores utilizam para avaliar o progresso e desempenho dos alunos em discussões online e a partir disso utilizar mineração de dados para avaliar o progresso dos alunos.

O presente trabalho se difere dos acima citados pois, apesar de se tratar de uma análise envolvendo jogos, o objeto da análise será a atividade no fórum do Warframe e não a atividade dos jogadores dentro do jogo. Também não será voltada para avaliar o desempenho dos jogadores com base em seus comentários nas postagens. Este trabalho busca investigar a relação entre a atividade no fórum e a quantidade média de jogadores ativos.

# 3. METODOLOGIA

Esta seção foi dividida em grandes partes. A Subseção 3.1 explica como a extração de dados foi realizada e todo o procedimento para o desenvolvimento do software responsável por isto. Já a subseção 3.2 é destinada a demonstrar como a análise dos dados foi realizada.

## 3.1. Extração dos Dados

O fórum do Warframe é dividido em 5 grandes categorias (*News, Community, Bug Reports, Feedback* e *International Forums*). Estes possuem subcategorias e algumas incluem fóruns e subfóruns, com postagens desde o ano de 2013, ano de lançamento do Warframe. Para este trabalho optou-se por selecionar a categoria *Feedback* para ser feita análise de dados, devido a sua grande interação entre os jogadores e os desenvolvedores do jogo. Depois de escolhida a categoria que seria trabalhada, selecionamos os dados que seriam salvos de cada postagem, que foram: Categoria (*Feedback*), Subcategoria (*General, Warframes, Weapons, Art, Animation & UI, Sound, Missions & Levels, Conclave*), Título da postagem, Data e hora da postagem, Plataforma a qual o jogador que comentou pertence (PC, *Playstation 4*, *Xbox One*), *nickname* do jogador, maestria do jogador (hierarquia de títulos que representam cada nível, indo do nível 1 ao 30, conforme vista na Tabela 1).

Tabela 1 - Níveis de Maestrias

| **Maestria** | **Nível** |
|---|---|
| Iniciante (*Initiate*) | 1 |
| Iniciante Prateado (*Silver Initiate*) | 2 |
| Iniciante Dourado (*Gold Initiate*) | 3 |
| Novato (*Novice*) | 4 |
| Novato Prateado (*Silver Novice*) | 5 |
| Novato Dourado (*Gold Novice*) | 6 |
| Discípulo (*Disciple*) | 7 |
| Discípulo Prateado (*Silver Disciple)* | 8 |
| Discípulo Dourado (*Gold Disciple*) | 9 |

| | |
|---|---|
| Predador (*Seeker*) | 10 |
| Predador Prateado (*Silver Seeker*) | 11 |
| Predador Dourado (*Gold Seeker*) | 12 |
| Caçador (*Hunter*) | 13 |
| Caçador Prateado (*Silver Hunter*) | 14 |
| Caçador Dourado(*Gold Hunter*) | 15 |
| Águia (*Eagle*) | 16 |
| Águia Prateada (*Silver Eagle*) | 17 |
| Águia Dourada (*Gold Eagle*) | 18 |
| Tigre (*Tiger*) | 19 |
| Tigre Prateado (*Silver Tiger*) | 20 |
| Tigre Dourado (*Gold Tiger*) | 21 |
| Dragão (*Dragon*) | 22 |
| Dragão Prateado (*Silver Dragon*) | 23 |
| Dragão Dourado (*Gold Dragon*) | 24 |
| Sábio (*Sage*) | 25 |
| Sábio Prateado (*Silver Sage*) | 26 |
| Sábio Dourado (*Gold Sage*) | 27 |
| Mestre (*Master*) | 28 |
| Mestre Intermediário (*Middle Master*) | 29 |
| Grão-Mestre (*Grand Master*) | 30 |

Fonte: Warframe (2013)

Para realizar a extração dos dados foi utilizada uma *spider*, classe do Scrapy feita para raspagem de dados em sites, onde é definida a URL de início, as páginas que devem ser seguidas e os dados a serem salvos. Foram utilizadas sete *spiders*, uma para cada subcategoria de Feedback, possuindo basicamente o mesmo comportamento, sendo modificada apenas a URL de início. O seguimento das páginas se dá seguindo a sequência: URL inicial da subcategoria escolhida, que possui várias páginas em sequência com todas as postagens; Seleção da URL de cada postagem; Seguimento das páginas seguintes, se houver, dos comentários da postagem, onde estão os dados a serem coletados. A Figura 3 ilustra como se dá o seguimento das páginas.

Figura 3 - Seguimento das páginas


Fonte: Autoria própria

A seleção dos dados é feita através das *tags* do HTML (*HyperText Markup Language*), e foi utilizado o *selector xpath* ou CSS (*Cascading Style Sheets*) para isso. Com a identificação do item no HTML, é necessário utilizar o *shell* do Scrapy para testar se a expressão do seletor retorna o dado que se está buscando. Funciona da seguinte maneira: fazemos a chamada do *shell* passando como parâmetro o site que se quer testar os seletores. Ao entrar no modo interativo, podemos utilizar a função *response* para testar as expressões. O retorno de cada chamada do *response* vai ser de acordo com as tags HTML que foram inseridas e passadas como parâmetro. A Figura 4 mostra a sequência para realizar o procedimento acima descrito, utilizando como exemplo o nickname do jogador.

Figura 4 - Sequência para extração dos dados


Fonte: Autoria própria

O primeiro seletor testado foi para fazer referência aos links das páginas que seriam seguidas. Tais referências foram passadas como parâmetros para as regras da *spider*. Segundo a *Scrapy 1.5 documentation,* regras são uma lista de um, ou mais, objetos do tipo *Rule*. Cada regra define como será o comportamento da *spider* ao rastrear o site, quais links devem ser seguidos ou não. Em *LinkExtractor* é passado como parâmetro a referência que indica como os links devem ser seguidos.

Outro seletor testado foi para os dados em si, para utilizar no código a expressão correta referente a cada dado a ser coletado. Com as expressões definidas, foi feita a função chamada *parse_item*. Nesta função definimos a forma como seriam extraídos os dados, os nomes para ser referência para os dados coletados e a verificação para próxima página de comentários. No Apêndice A é possível ver um exemplo de uma das *spiders* desenvolvida para este trabalho.

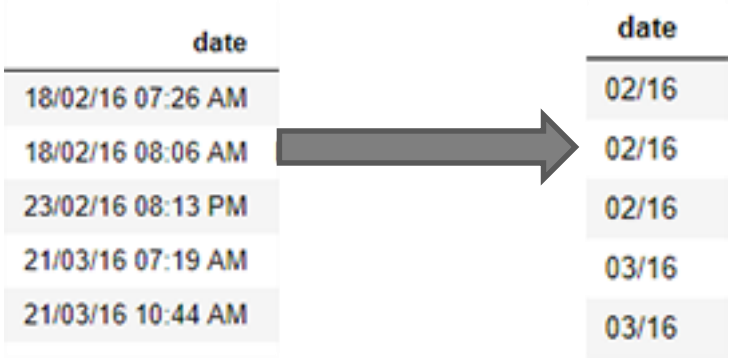
Após a definição das configurações das *spiders* elas foram executadas para realizar a coleta dos dados. Como cada subcategoria possui uma quantidade diferente de postagens e comentários, o tempo de execução de cada *spider* variou entre 2 e 20 horas, aproximadamente. Optou-se por salvar os dados no formato CSV (*Comma-Separated Values*), que é um formato de arquivo onde os dados armazenados são separados por vírgulas e podem ser visualizados em tabelas. Cada *spider* gerou um arquivo CSV referente a subcategoria que foi coletada.

Toda a evolução do projeto, codificação, bibliotecas necessárias, versões, estão disponíveis no repositório de código GitLab, e pode ser acessado através do link <https://gitlab.com/Rukiaski/tcc-project>.

### 3.2. Análise dos Dados

Seguindo os passos para mineração de dados, antes da análise é realizada a limpeza e pré-processamento nos dados coletados. Utilizando o Pandas, através do Jupyter Notebook, foi modificado o formato das datas, que passaram a conter apenas o mês e o ano referente ao comentário ao invés de conter dia, mês, ano e hora. Isto pode ser melhor observado na Figura 5, onde no lado esquerdo aparece o formato da data original, exatamente como foi extraída do fórum e no lado direito a data formatada.

Figura 5 - Ilustrando a modificação no formato da data




Fonte: Autoria própria

Uma outra etapa realizada no pré-processamento foi a substituição dos espaços entre as palavras pelo caractere *underscore*, para posteriormente trabalhar no Weka.

Com os dados devidamente processados, foi realizado o procedimento de redução, optando-se por selecionar apenas os dados referentes ao ano de 2017 para poder fazer a comparação os dados obtidos do Steam Charts. Foi gerada uma nova tabela contendo todas as postagens de todas as subcategorias referente ao ano de 2017, possuindo aproximadamente 280 mil linhas de conteúdo.

A primeira análise realizada foi a correlação entre a quantidade de postagens referente ao ano de 2017 e a quantidade média de jogadores. O coeficiente de correlação nos demonstra o grau de associação entre duas variáveis $x$ e $y$. Tendo como resultado um valor positivo, significa que os valores pequenos em $x$ tendem a corresponder aos valores pequenos em $y$, assim como os valores grandes em $x$ tendem a corresponder aos valores grandes em $y$. Se o resultado for negativo, significa que os valores pequenos em $x$ tendem a corresponder aos valores grandes em $y$, e os valores grandes em $x$ tendem a corresponder aos valores pequenos em $y$ (FREUND, 2006). A Equação 1 demonstra como é calculada a correlação entre duas variáveis. A variável x representa a quantidade de postagens que ocorreu em um mês e a variável y é a quantidade média de jogadores na plataforma PC. O cálculo é realizado para os meses do ano de 2017.

$$R = \frac{\sum xy - \frac{\sum x \times \sum y}{n}}{\sqrt{\left[\sum x^2 - \frac{(\sum x)^2}{n}\right] \times \left[\sum y^2 - \frac{(\sum y)^2}{n}\right]}}$$

Equação 1 - Equação da correlação.

Para fazer a outra parte da análise, foi escolhido o algoritmo *Apriori*, que mapeia os dados numa árvore coletando as informações e resultando regras de associação entre os dados (BAYARDO, 1998). As regras retornadas são divididas em duas partes separadas pelo conectivo lógico condicional. A primeira parte (premissa, a esquerda do conectivo lógico) é a condição estabelecida, e a segunda parte (consequência, a direita do conectivo lógico) é o que aparece quando a condição é verdadeira. Possuem como principais parâmetros suporte e confiabilidade. O suporte de uma regra significa a frequência com que ela aparece na base dados e a confiabilidade é a confiança que aquela regra retorna que é calculada dividindo a quantidade que o sucessor aparece pela quantidade que o antecessor aparece.

Para exemplificar melhor, tomemos como exemplos a Figura 6, onde podemos ver algumas regras retornadas pelo algoritmo *Apriori*. A primeira regra diz que se for o mês de outubro de 2017 (que possui uma frequência de 36866) então a plataforma do comentário vai ser PC (com frequência 29055), com uma confiabidade de 79%.

Figura 6 - Exemplos de regras de associação no Weka

```
Best rules found:

1. date=10_2017 36866 ==> platform=PC_Member 29055 <conf:(0.79)>
2. sub_category=general 147590 ==> platform=PC_Member 112641
<conf:(0.76)>
3. sub_category=warframes 57763 ==> platform=PC_Member 42583
<conf:(0.74)>
4. rank=Gold_Hunter 44201 ==> platform=PC_Member 32533 <conf:(0.74)>
5. platform=PC_Member 211883 ==> sub_category=general 112641
<conf:(0.53)>
```

Fonte - Weka

O Weka foi o programa escolhido para executar o algoritmo *Apriori*. Primeiro foi realizada uma conversão no arquivo CSV para o formato ARFF (*Attribute-Relation File Format*), que é um dos formatos que o Weka trabalha. Para executar no programa, optou-se por remover os dados referentes ao título da postagem e *nickname* do jogador. Em seguida, selecionou-se *Apriori* na aba *Associate*, onde possui os algoritmos de associação. Executamos alterando os valores das variáveis *lowerBoundMinSupport*, *upperBoundMinSupport* e *minMetric*.

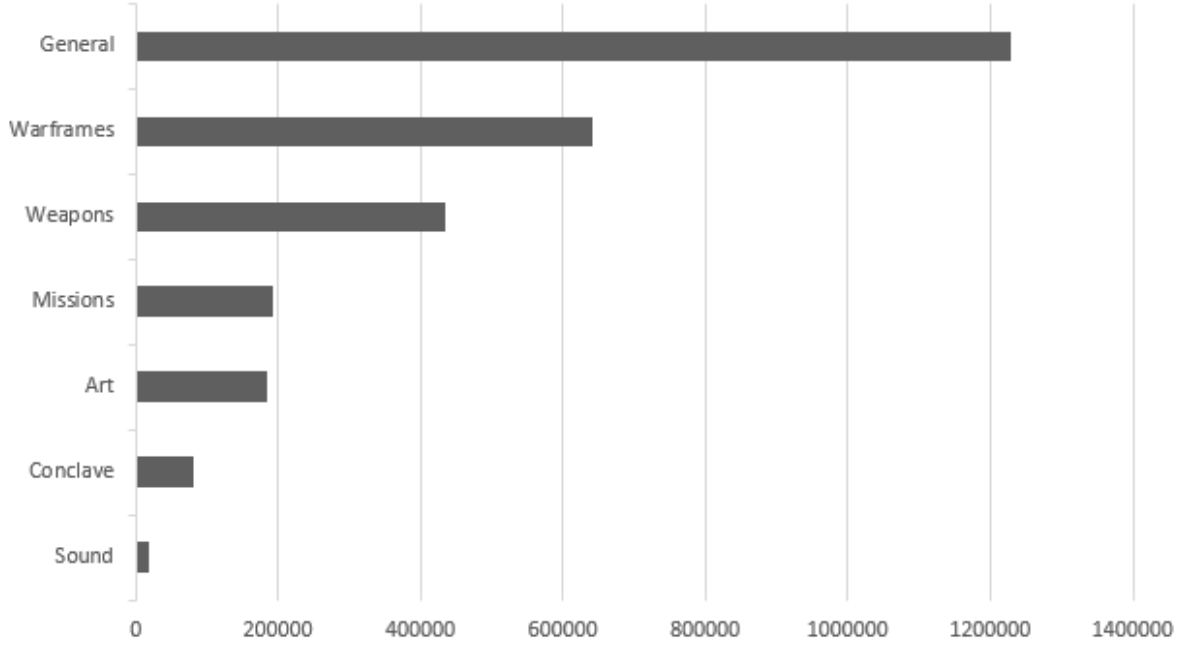
As variáveis *lowerBoundMinSupport* e *upperBoundMinSupport* são referentes ao suporte das regras na base de dados, sendo uma representando o mínimo e a outra o máximo, com intervalo definido entre 0,1 e 1, respectivamente, onde 0,1 equivale a 10% e 1 a 100%. Estes valores não foram alterados durante as execuções do algoritmo no Weka.

A outra variável *minMetric* é referente a confiança das regras retornadas. O valor inicial atribuído a esta variável foi 1, ou seja, retornaria regras que possuíssem uma confiança de 100%. A cada nova execução do algoritmo, o valor de minMetric foi sendo diminuído em 0,05 até chegar em 0,5.

# 4. RESULTADOS

Ao fim da execução de todas as *spiders*, obtivemos um total de aproximadamente 2,2 milhões de dados para se trabalhar. A subcategoria que contém mais dados é *General*, com aproximadamente 1,2 milhões de comentários. Já a categoria com menos dados é *Sound*, com aproximadamente 17 mil comentários. A Figura 7 ilustra em forma de gráfico a quantidade total de comentários de todas as subcategorias.

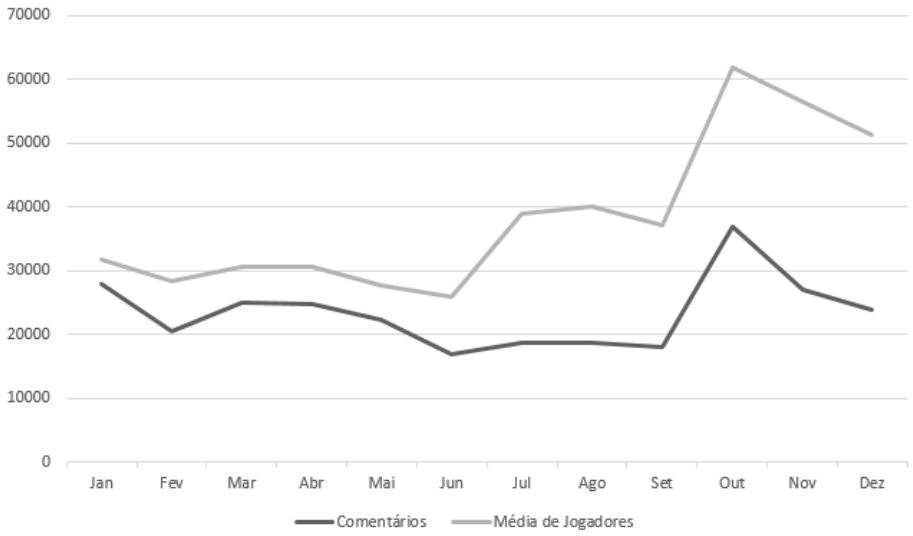
Figura 7 - Gráfico ilustrando a quantidade total de comentários.




Fonte: Autoria própria

A primeira análise realizada foi uma comparação entre a quantidade de comentários na categoria *Feedback* do fórum do *Warframe* no ano de 2017, que representam aproximadamente 10% do total de dados, com a quantidade média de jogadores da plataforma PC. Conforme pode ser observado no gráfico ilustrado na Figura 8, podemos notar que a quantidade de comentários no fórum está diretamente ligada à quantidade média de jogadores.

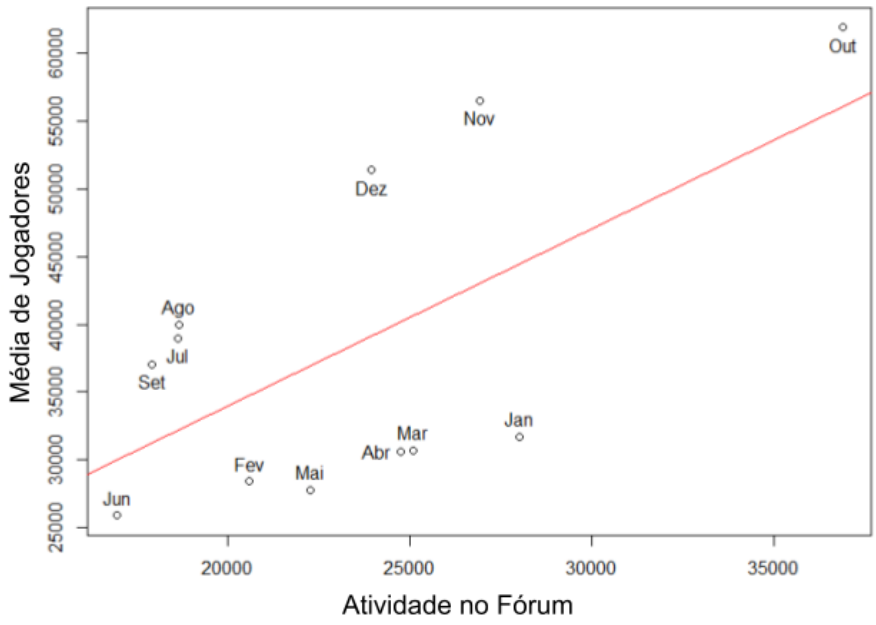
Figura 8 - Gráfico mostrando a quantidade de jogadores em relação a quantidade de comentários




Fonte: Autoria própria

Para comprovar essa relação entre as duas variáveis foi realizado o cálculo da correlação entre elas. A primeiro momento, a correlação entre elas não parece satisfatória, tendo em vista que o resultado do cálculo deu aproximadamente 37,14% de relação entre elas. Mas podemos observar na Figura 9 que a linha de tendência está aproximadamente no meio entre os dois semestres, onde abaixo da linha estão os valores referentes ao primeiro semestre e acima os valores referentes ao segundo semestre.

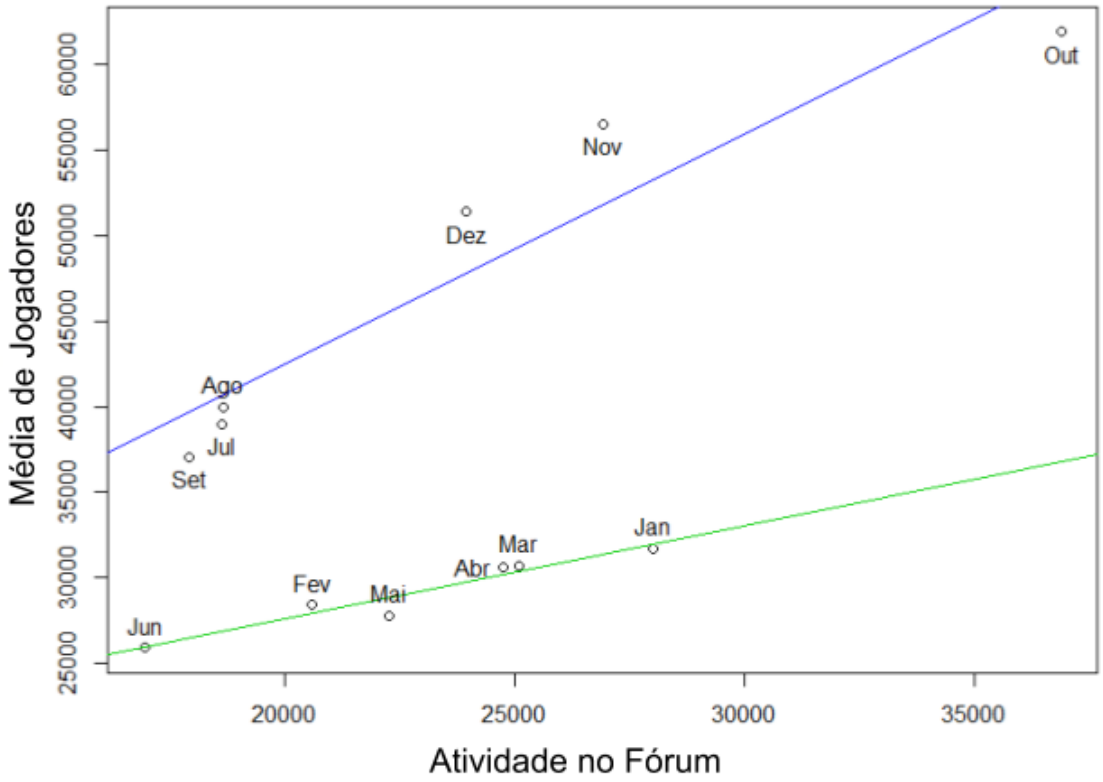
Figura 9 - Correlação entre todos os meses




Fonte: Autoria própria

Tendo isto em mente, podemos concluir que o semestre interfere na correlação entre essas duas variáveis. A partir disso, recalculamos a correlação entre essas variáveis, mas agora separando em dois semestres, de Janeiro a Junho e de Julho a Dezembro. Os resultados obtidos mostram o alto grau de correlação entre elas, pois o primeiro semestre retornou um resultado de aproximadamente 92% de correlação e o segundo 88%. A Figura 10 representa de forma gráfica esta análise.

Figura 10 - Correlação entre os semestres




Fonte: Autoria própria

A outra análise realizada utilizando o software Weka com o algoritmo *Apriori*, mesmo modificando os parâmetros, as regras retornadas foram poucas, conforme visto na Figura 6. As regras trouxeram padrões que já eram esperados, como por exemplo a maior atividade foi em outubro de 2017, quando foi lançada a expansão *Plains of Eidolon*, a maior quantidade de interações foi de jogadores da plataforma PC e as subcategorias que tiveram uma interação maior de jogadores foram *General* e *Warframes*.

# 5. CONCLUSÃO

Na parte da coleta dos dados o Scrapy demonstrou ser um framework efetivo para extração de dados web utilizando o código HTML das páginas, pois depois de ter sido programado corretamente, ele executou de forma rápida e precisa a coleta.

O tratamento dos dados realizado no Pandas, por meio do Jupyter Notebook, foi muito eficiente, pois mesmo a base de dados sendo relativamente grande, o processamento para o tratamento e filtragem nos dados eram executados de forma rápida.

A comparação entre a atividade no fórum e a quantidade média de jogadores revelou que quando há um aumento na média de jogadores na plataforma PC também há um aumento na atividade no fórum, o mesmo se aplicando quando há uma diminuição.

Os padrões retornados pelo Weka demonstraram algo que já era suposto, que a maior interação é de jogadores da plataforma PC, assim como a maior atividade no fórum foi no lançamento da expansão *Plains of Eidolon*.

## 5.1. Trabalhos Futuros

Como perspectiva para trabalhos futuros, pode ser realizada mineração dos dados com o Apriori novamente, mas utilizando-se outros parâmetros como filtro, como por exemplo remover as tuplas com jogadores da plataforma PC, ou remover meses em que o Warframe recebeu expansões, como ocorreu em outubro de 2017 com *Plains of Eidolon*. Pode ser feita também a execução de outros algoritmos de mineração de dados na base coletada.

Uma outra possibilidade com que o *crawler* possa coletar os dados de todo o fórum a partir da página inicial, coletando todas as categorias e subcategorias partindo desta página do fórum, para que a base de dados possa ser mais completa e precisa.

Pode-se também coletar os comentários dos jogadores e desenvolver um algoritmo para processar o texto em linguagem natural e salvar na base de dados os comentários em forma de tags para uma análise contextual.

Repetir o processo realizado de análise, mas com os dados dos anos anteriores e com dados coletados de outras categorias no fórum e comparar com os resultados obtidos neste trabalho.

Além disso é possível fazer outra análise de atividade no fórum em comparação com as outras plataformas em que o Warframe está disponível (PS4 e Xbox One) e analisar se a atividade de jogos semelhantes ao Warframe influencia na atividade do fórum.

## 6. REFERÊNCIAS


AGGARWAL, C. C. ***Data Mining:*** *The Textbook*. Nova York: Springer, 2015.

BAYARDO, R. J. *Efficiently Mining Long Patterns From Databases*. ***ACM Sigmod Record***. Vol. 27. No. 2. ACM, 1998.

DRINGUS, L. P.; ELLIS, T. Using data mining as a strategy for assessing asynchronous discussion forums. **Computers & Education**, [s.l.], v. 45, n. 1, p.141-160, ago. 2005. Elsevier BV. http://dx.doi.org/10.1016/j.compedu.2004.05.003.

EXTREMES, D. ***END USER LICENSE AGREEMENT*****.** Disponível em: <https://www.warframe.com/eula>. Acesso em: 11 out. 2018.

FAYYAD, U.; PIATETSKY-SHAPIRO, G.; SMYTH, P. From *Data Mining to Knowledge Discovery in Databases*. ***Ai Magazine***, Rhode Island, v. 17, n. 3, p.37-54, jul. 1996.

FRANK, E.; HALL, M. A.; WITTEN, I. H. ***The WEKA Workbench.*** Nova Zelândia: Morgan Kaufmann, 2016. Disponível em: <https://www.cs.waikato.ac.nz/ml/weka/Witten_et_al_2016_appendix.pdf>. Acesso em: 22 out. 2018.

FREUND, J. E. **Estatística Aplicada:** economia, administração e contabilidade. 11. ed. Porto Alegre: Bookman, 2006.

HAN, J.; KAMBER, M.; PIE, J. ***Data Mining:*** *Concepts and Techniques*. Waltham: Elsevier, 2012.

JUPYTER. ***The Jupyter Notebook 5.7.2 documentation.*** Disponível em: <https://jupyter-notebook.readthedocs.io/en/stable/index.html>. Acesso em: 17 out. 2018.

LAYTON, R. ***Learning Data Mining with Python.*** Birmingham: Packt Publishing, 2015. Disponível em: <https://www.packtpub.com/mapt/book/big_data_and_business_intelligence/9781784396053>. Acesso em: 09 out. 2018.

PANDAS. ***Pandas 0.23.4 documentation.*** Disponível em: <http://pandas.pydata.org/pandas-docs/stable/>. Acesso em: 15 out. 2018.

PINGEN, G. L. J. ***An implementation of a data-mining approach to strategy selection in Starcraft: Brood War*.**2014. 35 f. TCC (Graduação) - Curso de *Bachelor Of Science*, *Artificial Intelligence*, Radboud University Nijmegen, Alemanha, 2014.

PYTHON. ***Python 3.7.1 documentation*.** Disponível em: <https://docs.python.org/3/faq/general.html#what-is-python>. Acesso em: 07 out. 2018

ROGERS, S. ***Level Up!*** *The Guide to Great Video Game Design*. Estados Unidos: A John Wiley & Sons, 2010.

SCRAPY. ***Scrapy 1.5 documentation*.** Disponível em: <https://doc.scrapy.org/en/latest/intro/overview.html>. Acesso em: 14 out. 2018.

STATISTA. ***Value of the global video games market from 2012 to 2021 (in billion U.S. dollars).*** Disponível em: <https://www.statista.com/statistics/246888/value-of-the-global-video-game-market/>. Acesso em: 13 out. 2018.

SWEIGART, A. **Automatize Tarefas Maçantes com Python**: Programação Prática para Verdadeiros Iniciantes. São Paulo: Novatec, 2015.

WARFRAME. *Version* 24.0.8. Ontario, Canadá: Digital Extremes, 2013. Disponível em: <https://www.warframe.com/>. Acesso em: 11 out. 2017.

WEKA. Version 3.8.3. Nova Zelândia: The University Of Waikato, 2016. Disponível em: <https://www.cs.waikato.ac.nz/ml/weka/>. Acesso em: 21 mar. 2018.

# 7. APÊNDICES

## 7.1. Apêndice A

```
# -*- coding: utf-8 -*-
from typing import List

import scrapy
from scrapy.linkextractors import LinkExtractor
from scrapy.spiders import Rule, CrawlSpider
from warframe.items import WarframeItem


class FeedbackArtSpider(CrawlSpider):
    name = 'feedback_art'
    download_delay = 0.5
    allowed_domains = ['forums.warframe.com']

    start_urls = [
        'http://forums.warframe.com/forum/18-art-animation-ui/',
    ]

    rules = (
        Rule(
            LinkExtractor(restrict_css=['li.ipsPagination_next']),
            follow=True,
        ),
        Rule(
            LinkExtractor(restrict_css=['span.ipsType_break.ipsContained']),
            callback='parse_item',
        ),
    )

    def parse_item(self, response):
        for comment in response.css('article[id]'):
            yield {
                'category': 'feedback',
                'sub_category': 'art, animation and UI',
                'title': comment.xpath('//div/h1/span/span/text()').extract_first(),
                'date': comment.xpath('.//div[@class="ipsType_reset"]/a/time/@title').extract_first(),
                'platform': comment.xpath('.//li/span[contains(@style,"color")]/text()').extract_first(),
                'nickname': comment.xpath('.//a/span[contains(@style,"color")]/text()').extract_first(),
                'rank': comment.xpath('.//ul/li[@class="ipsType_break"]/text()').extract_first(),
            }

        next_page = response.css('li.ipsPagination_next a::attr(href)').extract_first()
        if next_page is not None:
            next_page = response.urljoin(next_page)
            yield scrapy.Request(next_page, callback=self.parse_item)
```