

Ferramenta para comparação de preços de itens do jogo Counter-Strike: Global Offensive

Leonardo Librelotto de Godoy¹, Henrique Gabriel Gularte Pereira¹

¹Curso de Ciência da Computação – Centro Universitário Franciscano
(UNIFRA) 97010-491 – Santa Maria – RS – Brasil

leogodoyllg@yahoo.com.br, henriquep@unifra.br

Abstract. *This paper consists of developing a price comparison tool for Counter-Strike: Global Offensive's items owned by the user and sold on Steam platform and OPSkins. Through the use of Steam Web API and OPSkins API was possible to query them to return information regarding the items in the user's inventory as well as their sales prices and average sales value. You can also search the items from any user with public inventory making the process of items exchange faster. With this tool the user will have the ability to analyze the price of his items and his friends in different markets, making it possible to choose the market with the best sale value.*

Resumo. *Este trabalho consiste em desenvolver uma ferramenta de comparação de preços de itens do jogo Counter-Strike: Global Offensive, pertencentes ao usuário e vendidos na plataforma Steam e OPSkins. Através do uso das APIs Steam Web e OPSkins foi possível realizar chamadas que retornaram informações referentes aos itens no inventário do usuário assim como seus respectivos valores de venda e valor médio de venda. Com essa ferramenta o usuário terá a capacidade de analisar o preço dos seus itens e de seus amigos em diferentes mercados, possibilitando a escolha do mercado com o melhor valor de venda.*

1. Introdução

O mercado de jogos eletrônicos tem tido um crescimento significativo nos últimos anos, em 2013 o mercado de jogos movimentou mais de 90 bilhões de dólares mundialmente e a tendência é que este número deve continuar aumentando a cada ano [THEESA 2014]. Em 2014, o gasto na indústria de jogos nos Estados Unidos da América foi superior as indústrias da música e filmes [Bryant 2016]. Um dos grandes responsáveis pelo crescimento no setor de jogos para computador é a plataforma de vendas Steam. Criada pela Valve, a Steam é a maior plataforma de distribuição de jogos para computador no mundo.

Desenvolvido pela criadora da Steam, Counter-Strike: Global Offensive (CS:GO) é um jogo online de tiro com perspectiva em primeira pessoa. Um dos fatores que o torna tão popular é um sistema de aparências que mudam o visual das armas, chamadas de *skins*, que são vendidas e trocadas entre os jogadores. Estima-se que centenas de milhares de itens do jogo são vendidos diariamente somente no mercado da plataforma.

Devido ao crescimento da comercialização de itens de jogos não somente no

mercado da Steam, mas principalmente com o surgimento de diversos sites de vendas, assim como o alto valor envolvido nessas transações, essa ferramenta se faz útil fornecendo ao usuário mais informações a respeito dos seus itens possibilitando a maximização do lucro e evitando perdas em negociações.

Esse artigo tem como objetivo descrever o processo de criação da ferramenta utilizada para a comparação de preços de vendas dos itens. A ferramenta foi desenvolvida usando a linguagem de programação PHP, também foi usado o framework Bootstrap, dando a ferramenta estrutura responsiva, ou seja, o conteúdo do site se adapta a tela do dispositivo usado para o acesso independentemente do tamanho da mesma, facilitando a visualização e a organização do conteúdo apresentado.

O trabalho está organizado em seções, sendo os objetivos apresentados nessa Seção, os trabalhos relacionados na Seção 2, o referencial teórico na Seção 3, abordando conceitos e tecnologias utilizadas no projeto. Na Seção 4 é apresentada a proposta do projeto. Na Seção 5 apresentada a metodologia de desenvolvimento escolhida e o processo de desenvolvimento na Seção 6 e por fim a conclusão e trabalhos futuros na Seção 7.

1.1 Objetivos Gerais

Desenvolver uma ferramenta que apresente ao usuário informações de forma precisa referentes aos valores de venda dos itens do jogo *Counter-Strike: Global Offensive* vendidos na plataforma Steam e no site OPSkins.

1.2 Objetivos Específicos

- Adquirir os dados do perfil do usuário Steam através da biblioteca OpenID;
- Buscar lista de amigos do usuário através da Steam Web API;
- Buscar o inventário de outros usuários através da Steam Web API;
- Adquirir os dados dos itens no inventário do usuário Steam através da Steam Web API;
- Buscar os valores de venda dos itens através da OPSkins API e Steam Web API;
- Realizar a filtragem de dados dos arquivos JSON adquiridos como resposta as chamadas às APIs;
- Opção de inspecionar itens no jogo.
- Apresentar os dados comparativos ao usuário.

2. Trabalhos Relacionados

Nesta seção serão apresentados trabalhos relacionados ao projeto desenvolvido. Esses trabalhos têm como objetivo auxiliar na elaboração do projeto.

Em “Buscapé: do empreendedorismo à inovação aberta” [Arruda et al. 2011], é descrito como a ferramenta Buscapé se tornou um dos principais sites comparadores de preços do Brasil. O artigo descreve sua fase inicial de desenvolvimento, as tecnologias criadas para o melhoramento do serviço e problemas encontrados para implantar um serviço deste tipo no país.

3. Referencial Teórico

Esta seção apresenta conceitos e tecnologias referente ao tema escolhido para a realização deste projeto.

3.1 Steam

Steam é uma plataforma de distribuição de conteúdo digital que oferece serviços de compra de jogos, filmes e softwares. Contando com mais de 125 milhões de usuários ativos em 2015 e acervo de milhares de jogos, Steam é a maior plataforma de distribuição de jogos do mundo [VG247 2015].

A plataforma foi criada em 2002 pela desenvolvedora de jogos Valve como uma forma de atualizar seus jogos automaticamente e com a possibilidade de implantar seu próprio sistema antipirataria. Com seu crescimento devido a acordos de distribuição com outras desenvolvedoras, a plataforma passou por diversas atualizações e implantações de serviços a seus usuários. Em 2007 foi lançada a Comunidade Steam, que agregou o aspecto de rede social a plataforma.

A plataforma possui um sistema de modificação de jogos que possibilita aos usuários criar, compartilhar e jogar seus jogos com modificações feitas pela comunidade, o Steam Workshop. Além da compra de jogos, os usuários podem negociar itens individuais que fazem parte de jogos específicos através do mercado Steam.

3.2 Counter-Strike: Global Offensive

Counter-Strike é um jogo online de tiro com visão em primeira pessoa desenvolvido pela Valve, teve sua primeira versão lançada em 1999 como uma modificação do jogo Half-Life. O título sofreu diversas atualizações durante os anos enquanto o número de jogadores continuava a crescer no mundo inteiro. Em 2012, foi lançada a versão Global Offensive, que rapidamente se tornou um dos jogos mais vendidos no mundo contando com dez milhões de jogadores únicos mensalmente [Counter-Strike 2016].

3.2.1 Sistema de Itens

O sistema de itens foi uma importante atualização que permitiu aos jogadores adquirirem itens de customização dos seus jogos favoritos e trocá-los com outros usuários ou negociá-los através do mercado da Steam. Como estes itens funcionam dentro do jogo depende de cada desenvolvedora, mas geralmente são opções de customização de aspectos visuais dos jogos.

3.3 PHP

PHP é uma linguagem de programação de código aberto independente de plataforma, com o objetivo principal de gerar conteúdo dinâmico para páginas na web, possibilitando a integração de código HTML no código PHP. Outra vantagem do uso da linguagem é o fato do código PHP ser executado diretamente no servidor sendo mostrado ao usuário somente o HTML resultante [Melo e Nascimento 2007].

PHP é uma das linguagens mais usadas para a criação de páginas dinâmicas na web. Se tratando de desenvolvimento de páginas dinâmicas, PHP tem excelente desempenho, é de fácil aprendizagem, integração com praticamente todos os bancos de

dados disponíveis, estabilidade e portabilidade. Tudo sem custo por ser uma linguagem de código aberto [Ullman 2012]. Devido a essas características que PHP é a linguagem utilizada na realização deste projeto.

3.4 HTML5

HTML5 (*Hypertext Markup Language*, versão 5) é uma linguagem utilizada para estruturar e apresentar conteúdos para a World Wide Web. Sua quinta versão foi lançada para atender as necessidades das atuais e futuras páginas web, trazendo melhoramentos nas suas funcionalidades de mídias como áudio e vídeo, semântica e acessibilidade.

Na raiz de uma página web está uma linguagem baseada em marcações, de fácil aprendizado e que qualquer dispositivo com um navegador possa lê-la: HTML [Castro e Hyslop 2012].

3.5 CSS

CSS (*Cascading Style Sheets*) é um padrão de desenvolvimento web cujo objetivo é formatar as informações entregues pelo HTML. Com o CSS é possível a modificação de fontes, fundos, cores, margens e divisões de páginas. Sua principal função é a separação entre conteúdo e formato de uma página [W3C 2016].

O CSS possibilita ao desenvolvedor criar uma página de estilos e ligar essa página às demais páginas de conteúdo, assim quaisquer alterações referentes a estilos e aparências não precisam ser feitas individualmente em cada página HTML do sistema e sim somente no arquivo CSS. Enquanto o código HTML define a estrutura do documento, o CSS define como o conteúdo é apresentado ao usuário final [Shaw 2015].

3.6 jQuery AJAX

jQuery é uma biblioteca JavaScript de código aberto com o objetivo de facilitar o uso de JavaScript. O jQuery foi desenvolvido para simplificar a manipulação de elementos e eventos, seleção de elementos DOM (*Document Object Model*) e desenvolver aplicações AJAX [w3schools 2016].

AJAX tem como objetivo tornar páginas web interativas com o usuário. O AJAX basicamente carrega os dados em segundo plano e os exibe na página sem a necessidade de atualizá-la. Com os métodos jQuery AJAX é possível requisitar texto, HTML e arquivos JSON de um servidor remoto e carregar os dados diretamente em um elemento HTML específico do site [w3schools 2016].

3.7 Bootstrap

Bootstrap é um framework de código aberto que facilita o desenvolvimento de páginas responsivas com tecnologia *mobile* minimizando o uso de CSS para tal tarefa, com isso reduzindo o tempo de desenvolvimento do projeto.

As principais vantagens do uso do Bootstrap são: funciona em todos os navegadores atuais; fácil customização; grande variedade de plugins customizáveis; Sistema responsivo [Niska 2014].

3.8 JSON

JSON (*JavaScript Object Notation*) é um formato de troca de dados computacionais baseado em um subconjunto da linguagem de programação, JavaScript porém não é necessária o uso da linguagem diretamente. JSON é um formato de texto independente de linguagem fazendo com que seja um formato ideal para troca de dados [JSON 2016].

JSON tem sido usado como uma alternativa ao XML pela facilidade das linguagens de programação em ler e interpretar o formato.

3.9 Steam Web API

Steam Web API é a API (*Application Programming Interface*) fornecida pela Valve para que desenvolvedores possam efetuar chamadas à Steam por informações e dados que podem ser usados nos seus próprios projetos. A API possibilita acesso a informações referente a perfis de usuário, acesso a itens no inventário, mercado da Steam e estatísticas globais da plataforma. Todas as chamadas à API feitas neste projeto terão como resposta dados no formato JSON [SteamWebAPI 2016].

As chamadas a essa API são feitas através de URLs (*Uniform Resource Locator*) onde são passados parâmetros específicos como nome e identificadores de itens e de usuários, chave de acesso a API e identificadores de classes de itens, tendo como retorno um arquivo JSON com informações que deverão ser filtradas posteriormente.

3.10 Biblioteca OpenID

OpenID é um sistema de identificação que permite o usuário efetuar login em diferentes sites sem a necessidade de criar uma nova conta em cada site visitado. Neste projeto, as informações de login do usuário serão enviadas diretamente a um provedor de identificação, Steam, onde foi feito todo o processo de validação [OpenID 2016]. Através da integração da Steam Web API com a biblioteca OpenID, os usuários podem realizar autenticação através das suas contas Steam sem que a ferramenta tenha acesso a seus dados de acesso.

3.11 OPSkins API

OPSkins é um mercado onde os usuários podem comprar e vender seus itens do jogo Counter-Strike: Global Offensive de forma segura e utilizando dinheiro real. Diferente do mercado da Steam onde o ganho das vendas é feito através de dinheiro virtual que só pode ser usado dentro da própria plataforma, ou seja, comprando jogos ou itens. OPSkins é o maior serviço de vendas de itens de jogos fora da Steam com cerca de cem mil itens vendidos diariamente [OPSkins 2016]. O site disponibiliza aos desenvolvedores sua própria API que dá acesso a informações referente aos itens vendidos no mercado. Todas as chamadas a essa API feitas neste projeto terão como resposta dados no formato JSON.

Assim como a API Steam Web as chamadas a essa API são feitas através de URLs onde são passados parâmetros específicos como nome de venda do item, chave de acesso a API e o código identificador de jogo, tendo como retorno um arquivo JSON com informações que deverão ser filtradas posteriormente.

4. Proposta

Este projeto tem por finalidade disponibilizar um sistema onde o usuário tenha acesso a um comparativo dos valores de mercado dos itens do jogo Counter-Strike: Global Offensive no seu inventário e no de seus amigos vendidos na Steam e no site OPSkins, assim como a média do valor de venda dos itens nos últimos sete dias.

Apresentando um comparativo com os preços dos itens vendidos nas duas plataformas almeja-se que o usuário tenha acesso a informações adicionais de seus itens, tornando o processo de pesquisa de preços mais rápido e apresentando ao usuário em qual plataforma ele terá maior lucro na negociação.

5. Metodologia

O Modelo utilizado para o desenvolvimento deste projeto foi o modelo em cascata ou também conhecido como ciclo de vida em cascata, pelo motivo de sugerir uma abordagem sequencial, favorecendo o desenvolvimento de um sistema linear [Pressman 2011]. Optou-se por este modelo, pois os requisitos eram bem definidos no início do projeto e após iniciado não ocorreriam mudanças radicais nas funcionalidades principais do sistema e eventuais pequenas mudanças não comprometeriam a estrutura do projeto.

No modelo cascata cada fase é encadeada a outra fase e ao final de cada fase consiste na geração de documentos aprovados [Sommerville 2007]. A fase seguinte só deve iniciar após a finalização da fase anterior conforme mostrado na Figura 1.

Neste projeto foi utilizado uma adaptação do modelo, onde foi definido os requisitos como mostrado nas Seções 5.1 e 5.2, o projeto do sistema como mostrado nos diagramas de casos de uso e de classe na Seção 5.3, e por fim, a implementação da ferramenta, mostrado na Seção 6.

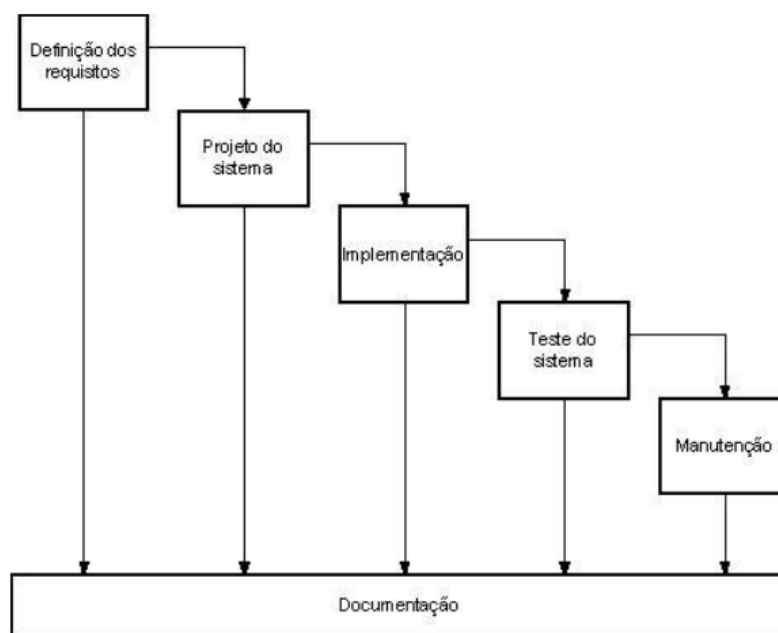


Figura 1 - Representação do Ciclo de Vida em Cascata [Pressman, 2011].

5.1 Requisitos Funcionais

Os requisitos funcionais (RF) descrevem as funcionalidades do sistema. É nos requisitos funcionais que são listados as entradas e saídas de dados e recursos fornecidos pelo sistema. Os RFs serão representados no diagrama de casos de uso mostrados na seção 5.3 [Sommerville 2007].

RF1 - Efetuar login da conta Steam.

RF2 - Buscar inventário do usuário.

RF3 - Buscar lista de amigos do usuário.

RF4 – Buscar inventário dos amigos.

RF5 - Apresentar os itens com seus respectivos valores de venda.

RF6 – Inspeccionar itens no jogo.

5.2 Requisitos Não Funcionais

Os requisitos não funcionais (RNF) são relacionados a propriedades do sistema como, confiabilidade e tempos de resposta, e não diretamente a funcionalidades do sistema. Os RNFs podem especificar desempenho, tecnologias necessárias, sendo importante não gerar restrições aos requisitos funcionais [Sommerville 2007].

RNF1 - Utilização de *session* através da biblioteca OpenID.

RNF2 - Linguagem de programação PHP/HTML5/CSS3.

5.3 Projeto do Sistema

Após análise e definição dos requisitos, que define as funcionalidades do sistema, iniciará o projeto e modelagem do sistema, onde é considerado como o sistema deverá funcionar internamente para que os requisitos possam ser atendidos [Pressman 2011].

Para o desenvolvimento da ferramenta foram utilizadas as linguagens PHP/HTML/CSS, assim como as APIs Steam Web e OPSkins, a biblioteca jQuery AJAX e o framework Bootstrap, dando ao sistema uma estrutura responsiva ao ser utilizado em dispositivos móveis.

A Figura 2 mostra o diagrama de casos de uso onde são apresentadas as funcionalidades do sistema e como elas interagem com o usuário final. Na Figura 3 é mostrado o diagrama de classe, onde é mostrado uma perspectiva mais aprofundada das funcionalidades do sistema a nível de programação.

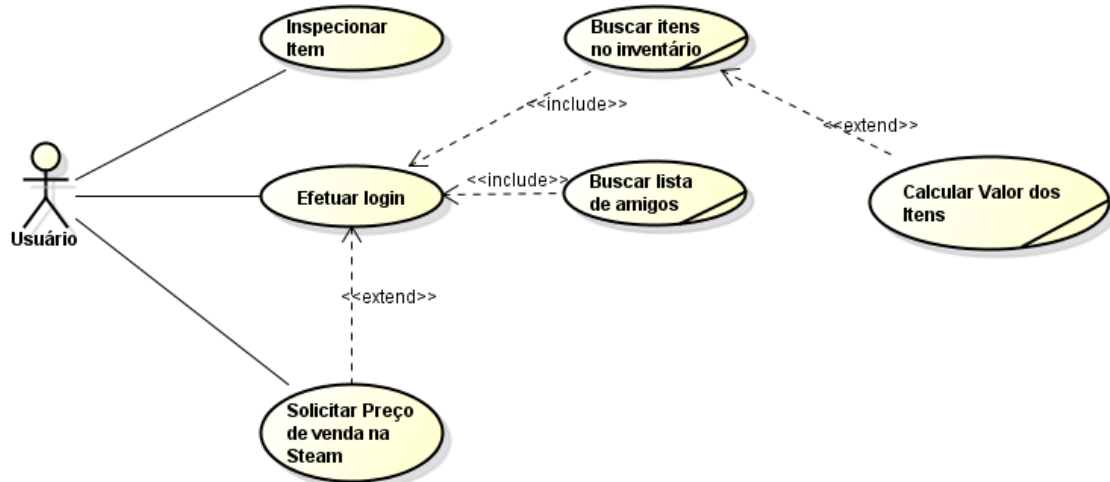


Figura 2 - Diagrama de Casos de Uso.

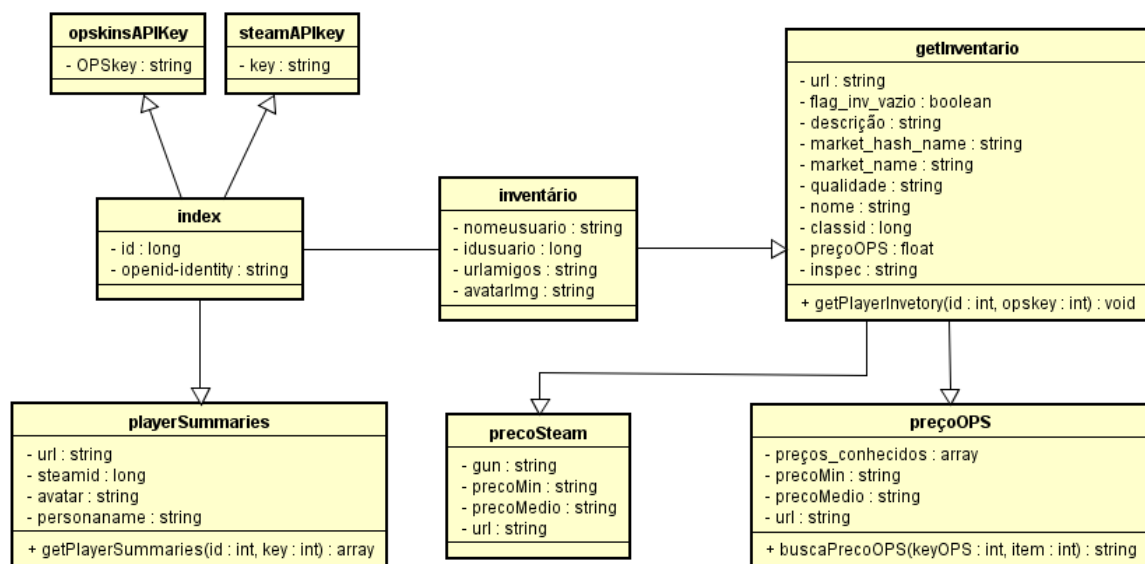


Figura 3 – Diagrama de Classe.

6. Desenvolvimento

Esta seção apresenta o processo de desenvolvimento da ferramenta e as funcionalidades do sistema.

6.1 Login com conta Steam através da API OpenID

Na Figura 4 é mostrada a interface inicial da ferramenta.

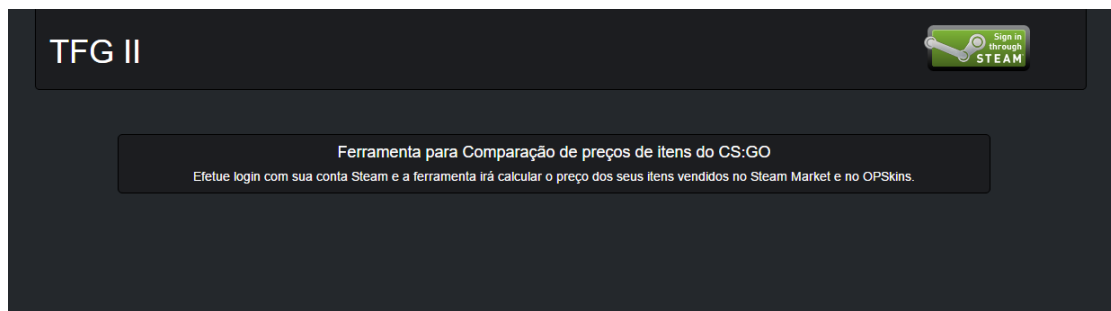


Figura 4 – Interface inicial.

Ao clicar no botão de login, a API OpenID irá acessar um arquivo XML da Steam contendo informações que levará ao redirecionamento para a página de autenticação da Steam, onde o usuário pode efetuar login de forma segura sem a ferramenta ter acesso aos seus dados privados, como mostrado na Figura 5.

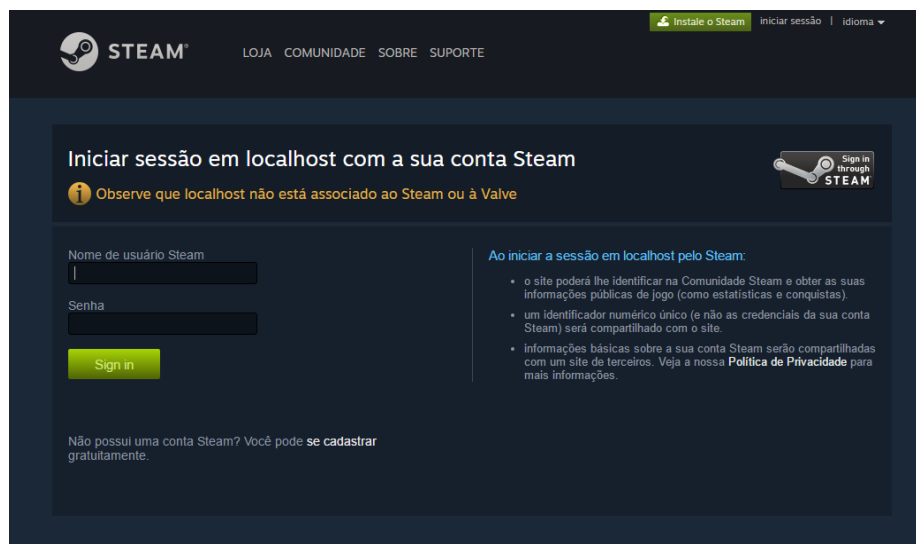


Figura 5 – Área de login.

6.2 Busca da lista de amigos

Na Figura 6 é mostrado um trecho de um arquivo JSON obtido como resposta a solicitação da lista de amigos do usuário.

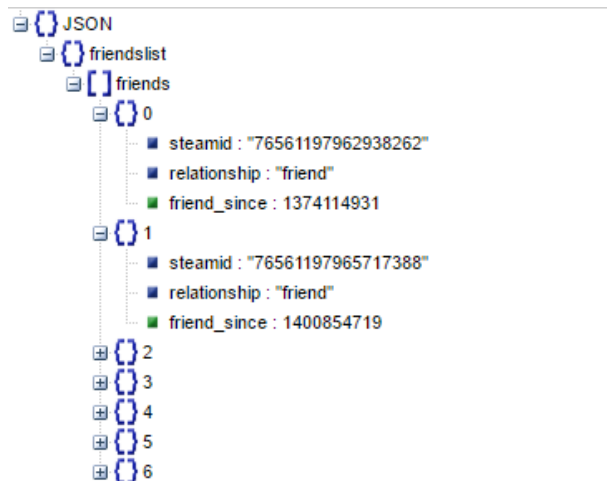


Figura 6 – Representação do resultado em JSON da lista de amigos.

Com o acesso ao *steamid* (identificador único gerado pela Steam para cada usuário) de cada membro na lista de amigos, é feita outra chamada à Steam Web API onde é extraído o *avatar* (imagem do perfil do usuário) e o *personaname* (nome utilizado pelo usuário no seu perfil) dos perfis de cada usuário na lista de amigos, como é mostrado na Figura 7.

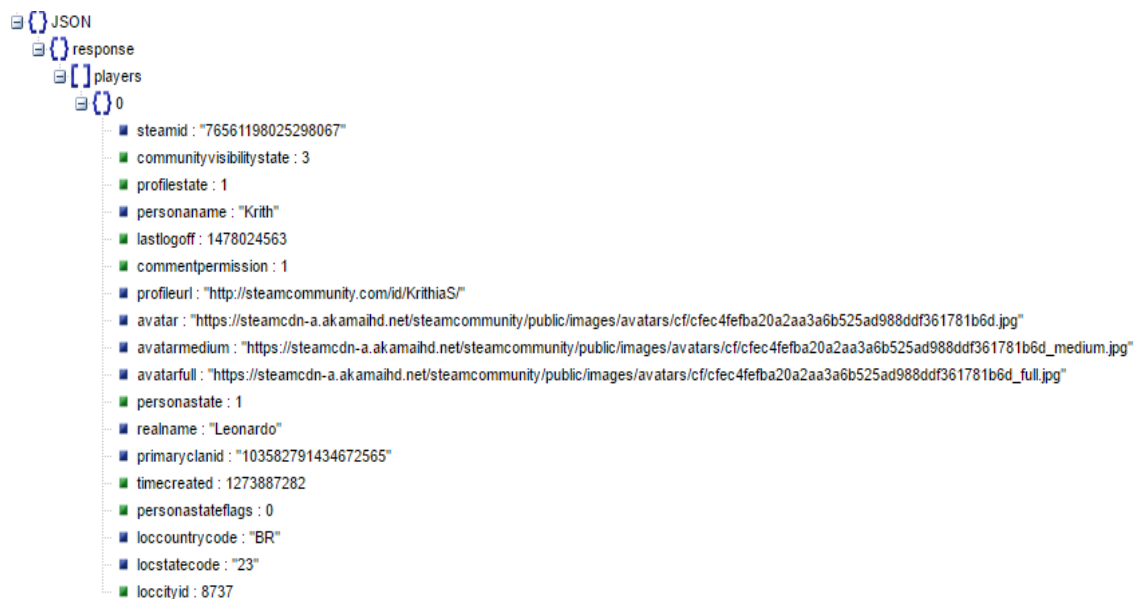


Figura 7 – Representação em JSON das informações do perfil do usuário.

6.3 Busca de itens no inventário

Os dados no arquivo JSON obtido como resposta à solicitação do inventário do usuário são divididos em *rgInventory* e *rgDescription*.

No *rgInventory* é encontrado o *id* (identificador único de um item) e o *classid* (identificador da classe da qual o item pertence) como mostra a Figura 8.

No *rgDescription* são encontrados os dados descritivos dos itens listados no *rgInventory* como *icon_url* (imagem do item), *name* (nome do item), *market_hash_name* (nome do item necessário para buscar o preço de venda na Steam), *market_name* (nome do item necessário para buscar o preço de venda no OPSkins) e informações de tipo, raridade, qualidade e ações externas, como é mostrado na Figura 9.

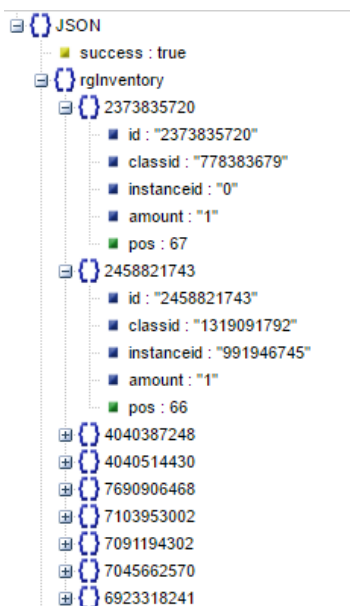


Figura 8 – Trecho de um arquivo JSON da busca de um inventário.

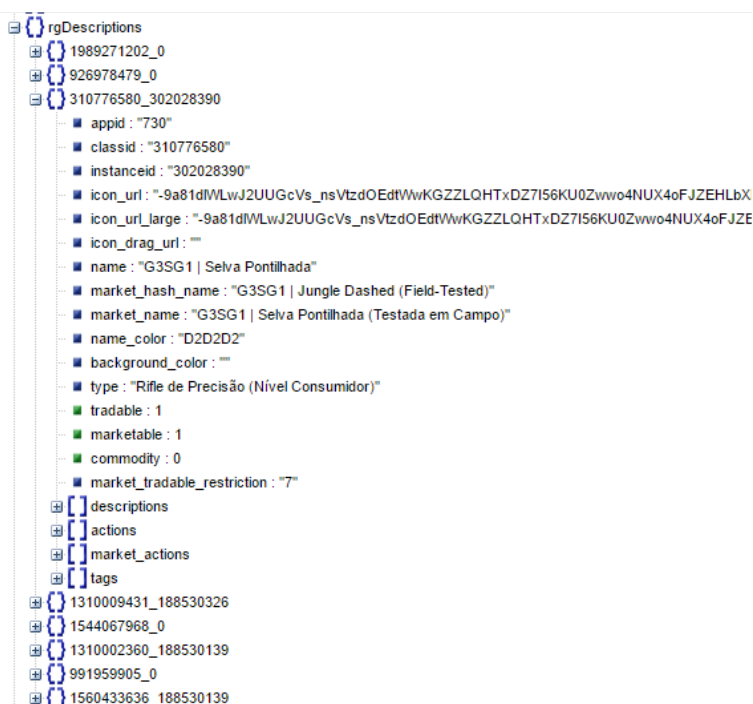


Figura 9 – Trecho de um arquivo JSON da busca de um inventário.

A Figura 10 mostra um trecho do código utilizado para percorrer o arquivo JSON e extrair dados do inventário.

```
function getPlayerInventory($id, $OPSkey){
    $url = "http://steamcommunity.com/profiles/" . $id . "/inventory/json/730/2";

    $json_objeto = file_get_contents($url);

    $json_decoded = json_decode($json_objeto);

    if (!isset($json_decoded->Error)){
        $flag_inv_vazio = True;
        foreach($json_decoded->rgInventory as $invID){
            foreach($json_decoded->rgDescriptions as $invDesc){
                if ($invDesc->classid == $invID->classid){
                    //Itens Negociáveis
                    if ($invDesc->marketable == 1){
                        $flag_inv_vazio = False;
                        $descricao = $invDesc->type;
                        $market_hash_name = $invDesc->market_hash_name;
                        foreach($invDesc->tags as $tags){
                            if (isset($tags->color)){
                                $qualidade = $tags->color;
                            }

                            if (strcmp($tags->category, "Exterior") == 0){
                                $descricao = $tags->name . " | " . $descricao;
                            }
                        }
                    }
                }
            }
        }

        $precoOPS = buscaPrecoOPS($OPSkey, $invDesc->market_name);
    }
}
```

Figura 10 – Trecho do código de busca de itens do inventário.

Após a filtragem dos dados são exibidos os itens ao usuário, como é mostrado na Figura 11. A cor da borda corresponde ao nível de raridade do item e a cor do nome do item corresponde a cor utilizada pela Steam para diferenciar itens especiais. Sob o nome do item é mostrado seu nível de qualidade e seu tipo.

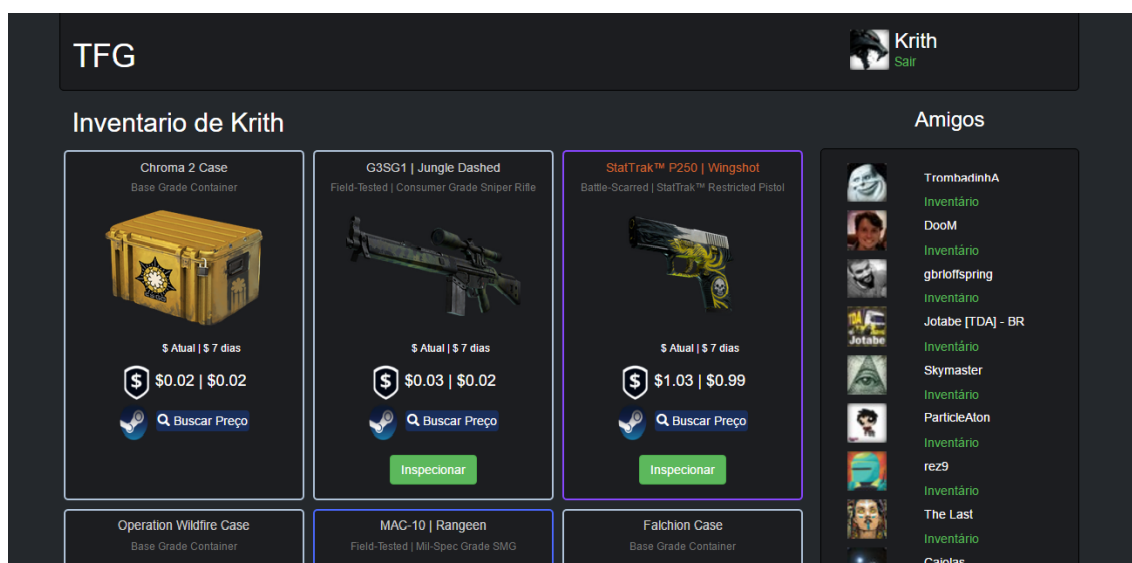


Figura 11 – Interface principal.

6.4 Busca de preços dos itens

Com o acesso ao *market_hash_name* do item, quando o botão “Buscar Preço” é clicado, é disparada uma ação feita com jQuery AJAX que envia à Steam Web API uma solicitação que tem como retorno um arquivo JSON contendo os campos *lowest_price* (menor preço do item sendo vendido no mercado) e *median_price* (preço médio de venda nos últimos sete dias), substituindo o botão pelos preços do item, como é mostrado na Figura 12.



Figura 12 – Processo de exibição do preço de um item vendido na Steam.

Com o acesso ao *market_name* do item são feitas duas solicitações à OPSkins API, onde é retornado o preço mínimo do item sendo vendido na plataforma no momento e o preço médio de venda do item nos últimos sete dias, posteriormente sendo exibidos ao usuário, como foi mostrado na Figura 12.

6.5 Inspeção do item

A Inspeção de um item permite ao usuário visualizar o item dentro do próprio jogo, contanto que o usuário possua o jogo instalado no dispositivo que a ferramenta está sendo acessada. Inspeccionar o item no jogo possibilita a visualização da real aparência do item, como sua qualidade e, dependendo do tipo do item, se a *skin* possui adesivos colados nela. Itens como caixas, chaves, kits musicais e cápsulas de adesivos não podem ser inspeccionados.

Ao clicar no botão “Inspeccionar” mostrado na Figura 12, se o usuário estiver com o jogo instalado no seu dispositivo a ferramenta solicitará a permissão para executar o jogo. Com a permissão concedida o jogo foi executado automaticamente e mostrado o item escolhido como mostra a Figura 13.

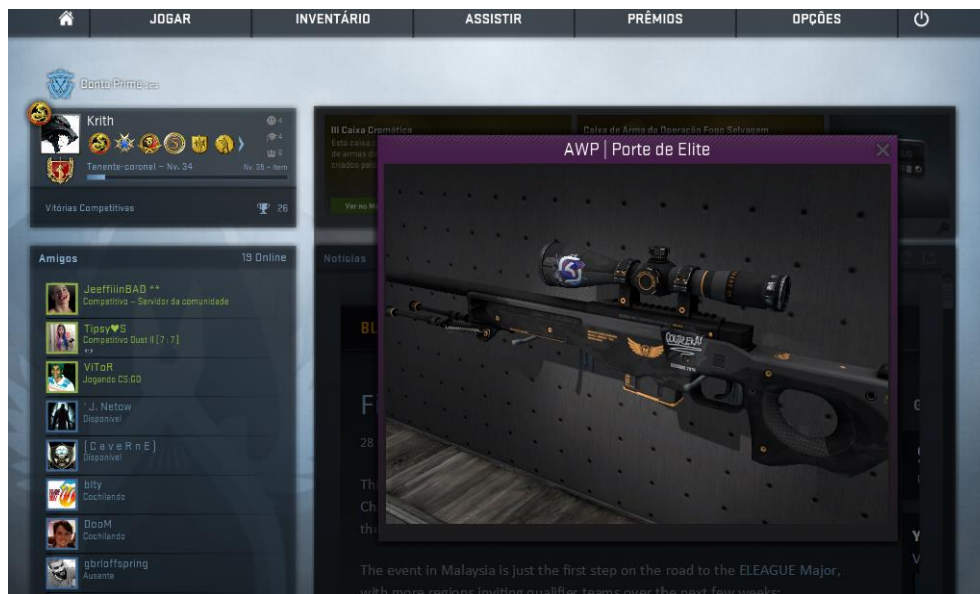


Figura 13 – Inspeção de um item no jogo.

7. Conclusões e Trabalhos Futuros

Durante o desenvolvimento do sistema, foi encontrado um problema no que diz respeito a aquisição dos preços de venda dos itens no mercado da Steam através da Steam Web API. A API possui um limite de requisições por segundo que podem ser realizadas. Sendo que inicialmente neste projeto as requisições solicitando o preço de um item na plataforma eram feitas assim que o item era encontrado no inventário, as chamadas à API ultrapassavam o limite de 1 requisição a cada 1,5 segundos, resultando no bloqueio temporário do IP, impedindo o envio de requisições por 24 horas.

Como alternativa para esse problema foi incluída a utilização de jQuery AJAX no projeto. Com o jQuery AJAX as requisições para buscar o preço do item no mercado da Steam são feitas através de um clique em um botão que envia a requisição através da API e retorna os preços para o determinado item, como foi mostrado na Seção 5.4.

O objetivo deste projeto foi alcançado ao disponibilizar a seus usuários uma forma de comparar os preços de seus itens em mercados distintos, que utilizam moedas distintas, em um só lugar.

Para trabalhos futuros, além do melhoramento no desempenho da ferramenta, poderá ser adicionado suporte a diferentes jogos além do *CS:GO* e abranger a pesquisa de preços a um número maior de plataformas de vendas, algo que não é feito por nenhum outro sistema.

8. Referências Bibliográficas

Arruda, Carlos. Rossi, Anderson. Penido, Erika (2011). “Buscapé: do empreendedorismo à inovação aberta”. Disponível em <<http://acervo.ci.fdc.org.br/AcervoDigital/Casos/Casos%202010/CF1005.pdf>>. Acessado em novembro de 2016.

Bryant, Yura (2016), “Getting Your Piece of the Growing Video Game Industry”. The

- Huffington Post. Disponível em <http://www.huffingtonpost.com/yura-bryant/getting-your-piece-of-the_b_8273850.html> Acessado em novembro de 2016.
- Castro, Elizabeth, Hyslop, Bruce. (2012), "HTML5 and CSS3". 7ª Ed. Berkeley, CA: Peachpit Press.
- Counter-Strike. Disponível em <<http://blog.counter-strike.net/index.php/about>> Acessado em maio de 2016.
- JSON. Disponível em <<http://www.json.org>> Acessado em maio de 2016.
- Melo, Alexandre A., Nascimento, Mauricio G. F. (2007), "PHP Profissional". Novatec.
- Niska, Christoffer (2014), "Extending Bootstrap". Birmingham, UK: Packt Publishing.
- OpenID. Disponível em <<http://openid.net/get-an-openid/what-is-openid/>> Acessado em maio de 2016.
- OPSkins. Disponível em <https://opskins.com/?loc=shop_help> Acessado em novembro de 2016.
- Pressman, R. S. (2011), "Engenharia de Software". 7ª ed. Rio de Janeiro: McGraw-Hill.
- Shaw, Peter (2015), "CSS3 Succinctly". Morrisville, NC: Sync.fusion.
- Sommerville, Ian (2007), "Engenharia de Software". 8ª ed. São Paulo: Addison Wesley.
- Steam Web API. Disponível em <<https://steamcommunity.com/dev?l=English>> Acessado em maio de 2016.
- THEESA (The Entertainment Software Association), "Games Economy". Disponível em <http://www.theesa.com/wp-content/uploads/2014/11/Games_Economy-11-4-14.pdf> Acessado em agosto de 2016.
- Ullman, Larry (2012), "PHP and MYSQL for Dynamic Web Sites". 4ª Ed. Berkeley, CA: Peachpit Press.
- VG247 (2015), "Steam has over 125 million active users, 8.9 concurrent peak". Disponível em <<https://www.vg247.com/2015/02/24/steam-has-over-125-million-active-users-8-9m-concurrent-peak/>> Acessado em abril de 2016.
- W3C Brasil. "CSS". Disponível em <<http://www.w3c.br/pub/Cursos/CursoCSS3/css-web.pdf>> Acessado em maio de 2016.
- w3schools "jQuery Introduction e jQuery – AJAX Introduction". Disponível em <http://www.w3schools.com/jquery/jquery_intro.asp> Acessado em setembro de 2016.