PONTIFÍCIA UNIVERSIDADE CATÓLICA DE GOIÁS

ESCOLA POLITÉCNICA E DE ARTES

GRADUAÇÃO EM ENGENHARIA DE COMPUTAÇÃO

Ferramenta de Educação em Segurança Digital: Um Jogo Interativo com Pygame

BRUNO RAPHAEL DA MATA SILVA BISPO

GOIÂNIA – GO,

2024

BRUNO RAPHAEL DA MATA SILVA BISPO

Ferramenta de Educação em Segurança Digital: Um Jogo Interativo com Pygame

Trabalho de Conclusão de Curso apresentado à Escola Politécnica e de Artes, da Pontifícia Universidade Católica de Goiás, como requisito parcial para a obtenção do título de bacharel em Engenharia de Computação.

Orientador: Ms. Fernando Gonçalves Abadia

GOIÂNIA – GO,

2024

BRUNO RAPHAEL DA MATA SILVA BISPO

Ferramenta de Educação em Segurança Digital: Um Jogo Interativo com Pygame

Este trabalho de Conclusão de Curso, julgado adequado para obtenção do título de Bacharel em Engenharia de Computação, e aprovado em sua forma final pela Escola Politécnica e de Artes da Pontifícia Universidade Católica de Goiás, em __/__/____.

_____

Prof. Me. Luiz Álvaro de Oliveira Júnior

Coordenador(a) de Trabalho de Conclusão de Curso

Banca examinadora:

_____

Prof. Me. Fernando Gonçalves Abadia

_____

Prof. Me. Rafael Leal Martins

_____

Prof. Me. Eugenio Júlio M. C. Carvalho

GOIÂNIA

12 de dezembro de 2024

AGRADECIMENTOS

Agradeço profundamente a todas as pessoas que, de maneira direta ou indireta, contribuíram para a realização deste trabalho.

Primeiramente, agradeço a Deus, pela força, sabedoria e coragem que me concedeu durante toda essa jornada. Sem Sua orientação e apoio, nada disso seria possível.

Sou imensamente grato aos meus pais, Alvino Bispo e Maria Zélia, pelo apoio incondicional, amor e incentivo ao longo de toda a minha jornada acadêmica. Vocês foram fundamentais para que eu chegasse até aqui, sempre acreditando em mim, mesmo nos momentos mais desafiadores. Ao meu irmão, Augusto Bispo, por sua paciência, compreensão e por sempre me motivar a seguir em frente.

Aos meus avós, Ivandes e Luzia, e à minha tia, Daniela, agradeço pelo carinho e apoio, mesmo que à distância. Suas palavras e gestos de incentivo, quando recebidos, sempre foram muito valiosos para mim.

Não posso deixar de agradecer aos meus amigos Henrique Barbosa, Jean Carlos e João Marcos, que me acompanharam em cada fase do processo, seja com conselhos, risadas ou simples gestos de amizade, que sempre tornaram os dias mais leves.

Agradeço imensamente ao meu orientador, Fernando Gonçalves e Rafael Leal Martins, por suas orientações competentes, paciência e dedicação. Suas visões críticas e o constante apoio foram essenciais para que este trabalho fosse possível. Sou grato pela oportunidade de aprender sob ambas tutorias e pela confiança que depositaram em meu trabalho.

A todos vocês, meu mais sincero agradecimento. Sem o apoio e presença de cada um, este TCC não teria sido possível.

RESUMO

Este trabalho tem como objetivo principal apresentar os conceitos e práticas de segurança para navegação segura na internet, utilizando a gamificação como ferramenta educacional. A crescente dependência da internet expõe os usuários a riscos cibernéticos, como golpes, fraudes e ataques. Neste contexto, propõe-se o desenvolvimento de um jogo educativo utilizando a biblioteca Pygame, para conscientizar sobre segurança digital de forma lúdica e interativa. A gamificação é utilizada para promover a participação ativa dos usuários, facilitando a compreensão de práticas essenciais de proteção contra ameaças online. O jogo busca ensinar tópicos como identificação de golpes, prevenção de ataques cibernéticos e segurança de dados, através de gráficos simplificados e experiências práticas. A metodologia aplicada abrange uma abordagem prática, na qual o Pygame é utilizado para criar experiências que combinam aprendizado e diversão, além de uma revisão bibliográfica para fundamentar os conceitos de segurança e gamificação. Assim, o trabalho visa não apenas introduzir conceitos teóricos de segurança digital, mas também demonstrar como a gamificação pode ser eficaz na mudança de comportamento dos usuários, tornando-os mais conscientes e preparados para lidar com riscos cibernéticos. Através da interatividade, espera-se capacitar os usuários a adotarem hábitos seguros no ambiente digital, contribuindo para a formação de uma cultura de segurança digital mais sólida e acessível.

Palavras-chave: Segurança da informação, Ataques cibernéticos, Golpes online, Pygame, Gamificação, Proteção de dados

ABSTRACT

This work aims to present the concepts and practices of security for safe internet browsing, using gamification as an educational tool. The increasing reliance on the internet exposes users to cyber risks such as scams, frauds, and attacks. In response to these challenges, an educational game was developed using the Pygame library to raise awareness about digital security in a playful and interactive manner. Gamification is used to promote active user participation, making it easier to understand essential practices for protection against online threats. The game aims to teach topics like identifying scams, preventing cyberattacks, and data security through simplified graphics and practical experiences. The applied methodology includes a practical approach where Pygame is used to create learning experiences that combine education and entertainment, as well as a literature review to support security and gamification concepts. Thus, this work seeks not only to introduce theoretical concepts of digital security but also to demonstrate how gamification can effectively change user behavior, making them more aware and prepared to deal with cyber risks. By leveraging interactivity, the game aims to empower users to adopt safe habits in the digital environment, contributing to the development of a stronger and more accessible digital security culture.

Keywords: Information security, Cyber attacks, Online scams, Pygame, Gamification, Data protection

LISTA DE ILUSTRAÇÕES

Figura 1 - Exemplo de ataque Sniffing 17
Figura 2 - Exemplo de ataque de força bruta 20
Figura 3 - Exemplo de ataque DDoS.....22
Figura 4 - Exemplo de configuração de recursos e janela no Pygame 31
Figura 5 - Exemplo de carregamento de imagem e configuração de fontes no Pygame 32
Figura 6 - Exemplo de responsividade no menu principal do Pygame.....33
Figura 7 - Configuração da grade de palavras cruzadas do jogo.....35
Figura 8 - Funções auxiliares para manipulação da grade de palavras cruzadas.....36
Figura 9 - Tela inicial do jogo 38
Figura 10 - Tela principal do jogo.....39
Figura 11 - Feedback visual para resposta correta no jogo 40
Figura 12 - Tela de dicas do jogo.....40

LISTA DE SIGLAS E ABREVIATURAS

CERT.BR - Centro de Estudos, Resposta e Tratamento de Incidentes de Segurança no Brasil.

CDNs - Content Delivery Network.

DoS - Denial of Service (Ataque de negação de serviço).

DDoS - Distributed Denial of Service (Ataque de negação de serviço distribuído).

HTTPS - Hypertext Transfer Protocol Secure.

IDS - Intrusion Detection System (Sistema de detecção de intrusão).

IPSec - Internet Protocol Security.

MFA - Multi-Factor Authentication (Autenticação multifator).

QoS - Quality of Service (Qualidade de Serviço).

SSL - Secure Sockets Layer.

TI - Tecnologia da Informação.

TLS - Transport Layer Security.

VPNs - Virtual Private Network.

VLAN - Virtual Local Area Network.

VM - Virtual Machine (Máquina Virtual).

Wi-Fi - Wireless Fidelity.

WAFs - Web Application Firewalls.

2FA - Autenticação de dois fatores.

SUMÁRIO

1. INTRODUÇÃO 11
   1.1. Justificativa 12
   1.2. Objetivo geral 12
   1.3. Objetivos específicos 12
   1.4. Metodologia 13
   1.5. Organização do Textual 14
2. REFERENCIAL TEÓRICO 15
   2.1. Ataques na Internet 15
   2.2. Interceptação de tráfego (Sniffing) 16
   2.3. Força Bruta (Brute Force) 18
   2.4. Negação de serviço (DoS e DDoS) 20
   2.5. Códigos maliciosos (Malware) 23
      2.5.1. Vírus 23
      2.5.2. Worms 24
   2.6. Gamificação 25
3. MATERIAIS E MÉTODOS 27
   3.1. Ambiente de Desenvolvimento 27
   3.2. Ferramentas Utilizadas 27
   3.3. Metodologia de Desenvolvimento 28
   3.4. Procedimentos Adotados 28
4. DESENVOLVIMENTO 30
   4.1. Estrutura do Código 30
   4.2. Arquivo Main.py 30
      4.2.1. Menu Inicial 32
      4.2.2. Atualização da Tela e Tratamento de Eventos 33

4.3. Arquivo Fase_facil.py 33

4.3.1. Importação e Inicialização 34

4.3.2. Configuração da Grade do Jogo 34

4.3.3. Lógica do Jogo 35

4.3.4. Tratamento de Eventos 36

4.4. Arquivo dicas.py 37

5. RESULTADOS 38

5.1. Resultado do desenvolvimento do jogo 38

5.1.1. Tela do Jogo 38

5.1.2. Tela de Dicas 40

5.2. Avaliação dos Testes com Usuários 41

5.3. Feedback dos Jogadores 42

5.4. Impacto Educacional 42

5.5. Limitações e Melhorias 43

6. CONSIDERAÇÕES FINAIS 44

REFERÊNCIAS BIBLIOGRÁFICAS 46

1. INTRODUÇÃO

A internet tornou-se uma ferramenta essencial no cotidiano, oferecendo acesso à informação, comunicação e entretenimento. No entanto, navegar na internet também implica riscos, como golpes, fraudes, ataques cibernéticos e acesso a conteúdo impróprio. A segurança da informação é um tópico cada vez mais relevante no mundo digitalmente conectado. Com o aumento da dependência das tecnologias da informação, a proteção de dados tornou-se uma prioridade tanto para indivíduos quanto para organizações. Enfrentamos uma luta constante contra-ataques cibernéticos e vulnerabilidades que ameaçam a integridade, confidencialidade e disponibilidade dos nossos dados (Shillair et al., 2015).

Os ataques cibernéticos representam uma ameaça significativa e crescente. Golpes e fraudes online são comuns e podem causar danos financeiros e emocionais severos às vítimas. Entre as fraudes mais frequentes estão golpes envolvendo prêmios falsos, fraudes em compras e vendas online, falsificação de identidade e roubo de informações pessoais. A educação e o conhecimento sobre medidas básicas de segurança na internet são o primeiro passo para alcançar uma navegação mais segura. A proliferação de malware e ransomware também é um risco substancial à segurança, podendo resultar em perda de dados e interrupção de serviços essenciais (Pušelj et al., 2019).

A conscientização sobre os riscos da internet é fundamental para a proteção dos usuários. Muitos usuários não têm o conhecimento necessário para mitigar ameaças cibernéticas, e programas educacionais podem desempenhar um papel vital na promoção de comportamentos seguros online (Ortega-Barón et al., 2021). A utilização de estratégias educacionais inovadoras, como a gamificação, pode tornar o aprendizado sobre segurança na internet mais atraente e acessível, especialmente para jovens e adolescentes, que são particularmente vulneráveis a esses riscos (Scholefield & Shepherd, 2019).

Nesse contexto, o Pygame, uma biblioteca open-source para desenvolvimento de jogos em Python, destaca-se como uma ferramenta eficaz para o desenvolvimento de jogos educativos voltados para a conscientização sobre segurança digital. O Pygame oferece uma interface simples e flexível para a criação de jogos em 2D,

utilizando bibliotecas gráficas como o SDL (Simple DirectMedia Layer). A documentação oficial do Pygame enfatiza a sua facilidade de uso e sua ampla aplicação em projetos educativos e experimentais, permitindo que até mesmo desenvolvedores iniciantes criem jogos e simulações interativas de forma rápida e eficiente (Pygame Documentation, 2023). Essa acessibilidade torna o Pygame uma ferramenta ideal para educadores que desejam gamificar o ensino de práticas de segurança digital, promovendo o aprendizado de maneira divertida e envolvente.

A seguir, os tópicos apresentarão a justificativa do estudo, os objetivos almejados e a organização textual dos demais capítulos.

1.1. Justificativa

A utilização da gamificação, aliada à interatividade proporcionada pelo Pygame, surge como uma estratégia eficaz para transformar o aprendizado sobre segurança digital, tornando-o mais envolvente e de fácil compreensão. Este trabalho justifica-se pela necessidade de capacitar os usuários a adotar práticas seguras no ambiente digital, especialmente ao abordar temas cruciais como a proteção contra golpes, fraudes online e a segurança de crianças e adolescentes na internet. Ao desenvolver um jogo educativo, espera-se proporcionar uma abordagem mais acessível e dinâmica, que permita aos usuários internalizar as melhores práticas de proteção digital de maneira eficaz e prazerosa.

1.2. Objetivo geral

O objetivo geral deste trabalho é apresentar os principais conceitos e dicas de segurança para navegar na internet de forma segura e responsável, utilizando a gamificação como meio para conscientizar os usuários.

1.3. Objetivos específicos

- Informar sobre os riscos de golpes e fraudes online e como identificá-los.
- Descrever medidas preventivas contra-ataques cibernéticos.
- Orientar sobre a segurança de crianças e adolescentes na internet.

1.4. Metodologia

Este trabalho foi desenvolvido com uma abordagem prática, utilizando a gamificação como ferramenta principal para promover a conscientização sobre segurança na internet. O desenvolvimento do projeto seguiu um caminho experimental, onde foi aplicado o Pygame, uma biblioteca de Python voltada para a criação de jogos 2D, como meio para engajar os usuários no aprendizado sobre segurança da informação.

A pesquisa iniciou-se com uma revisão bibliográfica, que incluiu a coleta de materiais acadêmicos, artigos e documentos técnicos sobre ataques cibernéticos, riscos de navegação na internet e o papel da gamificação em ambientes educacionais. Essa fase foi essencial para estabelecer a base teórica necessária para a criação do jogo educativo, identificando os principais desafios e medidas de segurança que deveriam ser abordados.

Com base nas informações coletadas, o projeto evoluiu para uma etapa prática, onde foi desenvolvido um jogo interativo com o objetivo de ensinar conceitos de segurança digital. A ferramenta utilizada foi o Pygame, devido à sua flexibilidade e facilidade de integração em ambientes educacionais. Durante o desenvolvimento, foram implementadas funcionalidades como dicas interativas, validação de respostas e progressão em fases, todas voltadas para reforçar o aprendizado sobre proteção online.

A metodologia aplicada seguiu um ciclo de desenvolvimento ágil, onde o protótipo do jogo foi testado e ajustado conforme o feedback obtido nas fases iniciais. A avaliação da eficácia do jogo foi baseada em testes com usuários, medindo o impacto da ferramenta na compreensão de conceitos-chave de segurança.

As áreas principais pesquisadas são:

- Segurança da Informação
- Cibersegurança e Ataques Cibernéticos
- Gamificação e Educação Digital
- Desenvolvimento de Jogos com Pygame
- Técnicas e conceitos para a escrita correta de um Trabalho de Conclusão de Curso (TCC)

1.5. Organização do Textual

Este trabalho está organizado em seis capítulos, abrangendo desde a fundamentação teórica até as conclusões e resultados.

O capítulo 1 oferece uma introdução ao tema, destacando a relevância da segurança da informação e a necessidade de métodos inovadores para conscientizar os usuários sobre proteção online. Também são apresentados os objetivos do projeto e a importância da gamificação como ferramenta educacional.

O capítulo 2 aprofunda o referencial teórico, abordando os principais conceitos relacionados à segurança digital, incluindo ataques cibernéticos como malware, phishing e DDoS, além de estratégias de mitigação de riscos. O conceito de gamificação também é explorado, destacando seu potencial para aumentar o engajamento dos usuários no aprendizado sobre segurança.

No capítulo 3, são descritos os métodos utilizados no desenvolvimento do jogo educativo, desde a pesquisa inicial até a implementação das funcionalidades com Pygame. São detalhados os critérios para a escolha dos conteúdos de segurança e a maneira como foram integrados ao jogo, assim como as etapas de desenvolvimento das interações.

O capítulo 4 discute o desenvolvimento do jogo, incluindo a criação das fases e a lógica de validação das respostas dos jogadores. No capítulo 5, são apresentados os resultados obtidos nos testes com usuários, analisando o impacto do jogo na assimilação dos conceitos de segurança. O capítulo 6 finaliza com as considerações finais, destacando os objetivos alcançados, as limitações do projeto e possíveis melhorias para trabalhos futuros.

2. REFERENCIAL TEÓRICO

Este capítulo abordará os principais tipos de ataques cibernéticos, suas características e as medidas de proteção. Serão descritos os ataques na internet, como malware, phishing e negação de serviço (DoS e DDoS), que ameaçam a segurança da informação, detalhando seus mecanismos e impactos. Também serão explorados ataques específicos, como interceptação de tráfego (Sniffing) e força bruta (Brute Force), que visam roubar informações confidenciais ou quebrar senhas. Além disso, o capítulo trata de códigos maliciosos, incluindo vírus, worms e outras formas de malware, ressaltando a importância de medidas preventivas e o uso de ferramentas de segurança para mitigar esses riscos.

2.1. Ataques na Internet

Os ataques na Internet representam uma ameaça significativa para a segurança da informação. Estes ataques podem assumir diversas formas e são realizados por indivíduos mal-intencionados, grupos organizados e até mesmo estados-nação. Um dos tipos mais comuns de ataques é o malware, software malicioso projetado para infectar sistemas e causar danos. Isso pode incluir ransomware, que criptografa os dados da vítima e exige um resgate para restaurá-los, e spyware, que coleta informações confidenciais sem o conhecimento do usuário (CARTILHA, 2012).

Além disso, os ataques de phishing são outra ameaça prevalente. Nesses ataques, os criminosos se passam por entidades confiáveis, como bancos ou empresas, e tentam enganar os usuários para que divulguem informações confidenciais, como senhas ou números de cartão de crédito. Os ataques de negação de serviço (DDoS) também são comuns, visando sobrecarregar servidores e redes, tornando os serviços inacessíveis para os usuários legítimos (CARTILHA, 2012).

Além dos ataques diretos, às vulnerabilidades nos sistemas de TI representam outra grande preocupação para a segurança da informação. Uma vulnerabilidade é uma fraqueza em um sistema que pode ser explorada por um atacante para comprometer a segurança desse sistema. Essas vulnerabilidades podem surgir

devido a falhas de projeto, erros de programação ou falta de atualizações de segurança (CARTILHA, 2012).

Por exemplo, uma vulnerabilidade de software pode permitir que um invasor execute código malicioso em um sistema, ganhando assim acesso não autorizado. Da mesma forma, uma configuração inadequada de segurança pode deixar portas abertas para ataques, tornando os sistemas mais suscetíveis a invasões (CARTILHA, 2012).

2.2. Interceptação de tráfego (Sniffing)

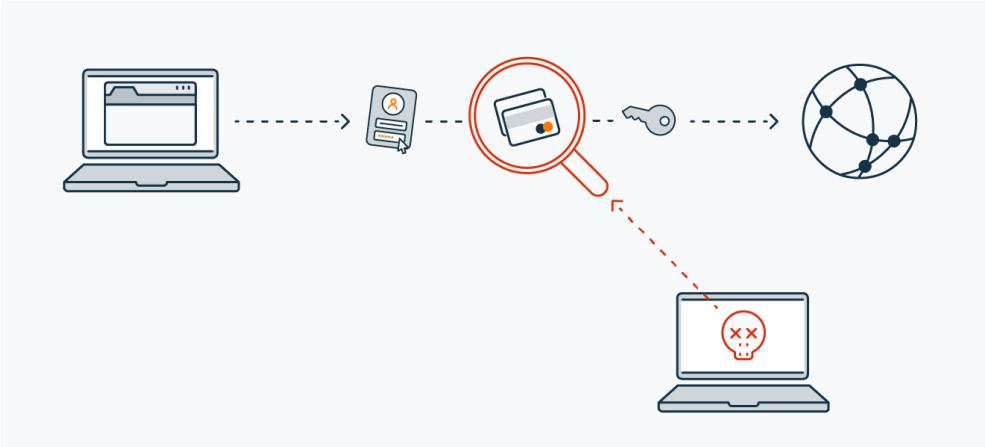
O ataque cibernético conhecido como Sniffing representa uma ameaça significativa à segurança da informação, permitindo que invasores capturem dados sensíveis transmitidos em redes. Conforme destacado na Cartilha de Segurança para Internet, o Sniffing é uma técnica utilizada para interceptar e analisar o tráfego de rede, extraindo informações confidenciais sem o conhecimento dos usuários. Essa prática maliciosa coloca em risco a privacidade e a integridade dos dados, exigindo medidas eficazes de proteção.

O Sniffing pode ser realizado de diversas formas, desde o monitoramento passivo do tráfego de rede até a utilização de softwares especializados para capturar pacotes de dados. Segundo a Cartilha (2012) essa técnica Sniffing pode ser utilizada de duas maneiras, ou seja, de forma legítima e de forma maliciosa.

Legítima: por administradores de redes, para detectar problemas, analisar desempenho e monitorar atividades maliciosas relativas aos computadores ou redes por eles administrados (CARTILHA, 2012).

Maliciosa: por atacantes, para capturar informações sensíveis, como senhas, números de cartões de crédito e o conteúdo de arquivos confidenciais que estejam trafegando por meio de conexões inseguras, ou seja, sem criptografia (CARTILHA, 2012).

Figura 1 - Exemplo de ataque Sniffing



Fonte: AVAST (2023).

Para proteger redes e dados contra ataques de sniffing, diversas ferramentas e técnicas são amplamente utilizadas, com foco na defesa e mitigação de vulnerabilidades. Essas medidas visam garantir a segurança da informação ao dificultar a interceptação e análise de tráfego de dados por agentes maliciosos.

Uma das principais ferramentas de defesa é a criptografia de dados, que assegura que as informações transmitidas na rede estejam codificadas, tornando sua interpretação impossível sem a chave de descriptografia correta. Protocolos como o HTTPS (Hypertext Transfer Protocol Secure), SSL/TLS (Secure Sockets Layer/Transport Layer Security) e o IPSec (Internet Protocol Security) são amplamente adotados para proteger a comunicação de dados em redes públicas e privadas. A criptografia ponta a ponta em aplicativos de mensagens também é uma medida eficaz que impede que terceiros acessem o conteúdo das comunicações. Stallings (2017) destaca que a criptografia é essencial para garantir a confidencialidade e integridade das informações transmitidas, sendo uma das bases mais importantes da segurança digital.

Outra solução fundamental é a implementação de redes seguras com autenticação robusta, como o uso de redes privadas virtuais (VPNs). As VPNs criam túneis criptografados para o tráfego de rede, dificultando que sniffers acessem os dados transmitidos. Além disso, a autenticação multifator, como o uso de autenticação de dois fatores (2FA), e a adoção de senhas fortes aumentam a segurança, reduzindo o risco de credenciais comprometidas. Tanenbaum e Wetherall (2011) apontam que mecanismos como VPNs e autenticação forte são pilares na proteção contra

interceptações de tráfego.

Ferramentas de monitoramento de tráfego de rede, como Wireshark e tcpdump, também desempenham um papel crucial. Quando utilizadas de forma legítima, essas ferramentas ajudam administradores a identificar padrões anômalos e possíveis ataques, permitindo a adoção de medidas preventivas. Além disso, a configuração adequada de dispositivos de rede, como switches, é essencial para dificultar a execução de ataques de sniffing. Técnicas como Port Security e VLAN Segmentation, recomendadas por Cisco Systems (2021), auxiliam na segmentação e proteção do tráfego, reduzindo a superfície de ataque.

Por fim, a conscientização dos usuários é uma linha de defesa indispensável. Treinamentos regulares sobre os perigos de redes Wi-Fi públicas inseguras e o uso de práticas seguras de navegação podem mitigar significativamente os riscos associados ao sniffing. Whitman e Mattord (2018) enfatizam que a educação em segurança da informação é uma das estratégias mais eficazes para prevenir ataques cibernéticos. A combinação de ferramentas tecnológicas robustas com boas práticas de segurança constitui, portanto, uma barreira eficiente contra sniffing e outras ameaças cibernéticas.

2.3. Força Bruta (Brute Force)

Ataques de força bruta são um método comum usado por criminosos cibernéticos para obter acesso não autorizado a sistemas e contas. Esses ataques envolvem o uso de ferramentas automatizadas para tentar diferentes combinações de nomes de usuário, senhas ou chaves de criptografia até que a combinação correta seja encontrada. De acordo com a Equipe de Resposta de Emergência em Informática do Brasil (CERT.BR) em sua Cartilha de Segurança para Internet, ataques de força bruta podem ser usados para obter acesso a sistemas, roubar informações confidenciais e interromper serviços (CERT.BR, 2022).

Ataques de força bruta podem ser divididos em duas categorias ataques online e ataques offline. Ataques online são realizados diretamente contra o sistema alvo, enquanto ataques offline são realizados em uma cópia dos dados criptografados (CERT.BR, 2022). Ataques online são menos eficientes do que ataques offline porque estão limitados pela resposta do sistema alvo. No entanto, eles ainda podem ser eficazes se o atacante tiver tempo e recursos suficientes.

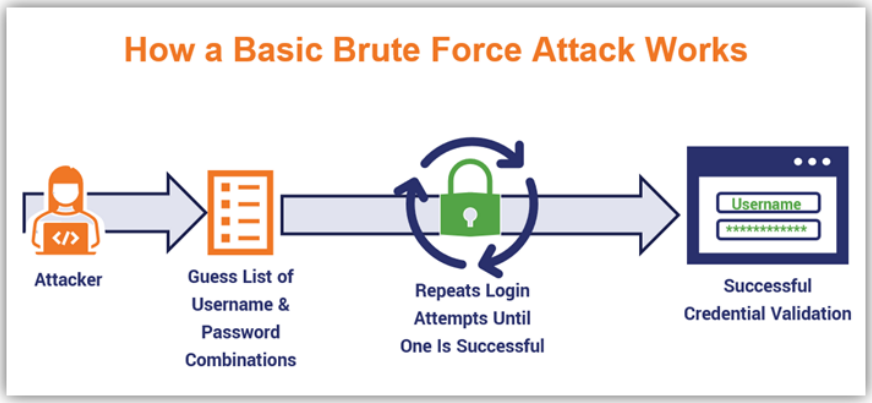
Ataques offline, por outro lado, podem ser muito mais eficazes porque podem ser realizados em uma taxa mais rápida do que ataques online. Em ataques offline, o atacante obtém uma cópia dos dados criptografados e, em seguida, usa um computador poderoso para tentar diferentes combinações de chaves de criptografia até que a combinação correta seja encontrada. De acordo com a CERT.BR, ataques offline podem ser usados para quebrar senhas, decifrar comunicações criptografadas e roubar informações confidenciais (CERT.BR, 2022).

Para se proteger contra-ataques de força bruta, a CERT.BR recomenda o uso de senhas fortes, a implementação de políticas de bloqueio de contas e o uso de autenticação multi-fator (MFA) (CERT.BR, 2022). Senhas fortes devem ter pelo menos 12 caracteres e incluir uma combinação de letras maiúsculas e minúsculas, números e caracteres especiais. As políticas de bloqueio de contas devem ser implementadas para bloquear contas após um certo número de tentativas de login fracassadas. A MFA exige que os usuários forneçam duas ou mais formas de autenticação, como uma senha e uma impressão digital ou uma senha e um código de tempo único enviado para um dispositivo móvel.

Além dessas medidas, a CERT.BR recomenda o uso de criptografia para proteger dados confidenciais e o uso de sistemas de detecção de intrusão (IDS) para detectar e prevenir ataques de força bruta (CERT.BR, 2022). A criptografia pode ser usada para proteger dados em repouso e em trânsito, enquanto os IDS podem ser usados para detectar tentativas de login incomuns ou padrões de atividade que possam indicar um ataque de força bruta.

Em conclusão, ataques de força bruta representam uma ameaça séria à segurança de sistemas e contas. Através do uso de senhas fortes, da implementação de políticas de bloqueio de contas e da utilização de MFA, as organizações podem reduzir o risco de ataques de força bruta bem-sucedidos. Além disso, o uso de criptografia e IDS pode ajudar a detectar e prevenir ataques de força bruta. Como observa a CERT.BR, a segurança é um processo contínuo que requer atenção e adaptação constantes a novas ameaças e vulnerabilidades (CERT.BR, 2022).

Figura 2 - Exemplo de ataque de força bruta



Fonte: THE SSL STORE (2023).

2.4. Negação de serviço (DoS e DDoS)

Os ataques de negação de serviço (DoS) e distribuídos (DDoS) são crescentes ameaças à segurança cibernética que visam interromper os serviços de um host conectado à internet, sobrecarregando-o com tráfego malicioso. A Cartilha de Segurança para Internet – Versão 4.0 descreve o DoS como um ataque em que criminosos interrompem os serviços de um host enviando um fluxo constante de tráfego, como solicitações falsas, sobrecarregando o sistema e impedindo que ele processe o tráfego legítimo.

No entanto, o DDoS é uma forma mais sofisticada de ataque, envolvendo um grupo de dispositivos conectados à internet, às vezes em escala global, que inundam o servidor de destino com solicitações fraudulentas de várias fontes ao mesmo tempo. Esses ataques são cada vez mais comuns e difíceis de mitigar, podendo ser

executados por um longo período de tempo.

A detecção de ataques de negação de serviço é fundamental no gerenciamento de sistemas tolerantes a falhas, pois anomalias no tráfego de rede podem afetar a disponibilidade e a qualidade do serviço (QoS). Sistemas detectores de intrusão em redes de computadores são utilizados para analisar o tráfego de rede com o objetivo de detectar ataques. A análise baseada em anomalias permite detectar ataques através da análise do comportamento do tráfego de rede.

A Microsoft destaca que os DDoS são algumas das ameaças cibernéticas mais comuns e podem comprometer negócios, segurança online, vendas e reputação. É possível que hackers também possam infiltrar seu banco de dados durante um ataque, acessando informações confidenciais.

Para se proteger contra ataques DoS e DDoS, é essencial adotar um conjunto abrangente de medidas proativas e reativas. A implementação de soluções específicas de mitigação de DDoS, como serviços de proteção em nuvem, é altamente recomendada. Provedores especializados, como Cloudflare, Akamai e AWS Shield, oferecem serviços capazes de filtrar tráfego malicioso antes que ele alcance os servidores, garantindo a disponibilidade dos sistemas legítimos. Esses serviços são considerados ferramentas eficazes na mitigação de ataques de volumetria, conforme destacado por Stallings (2017).

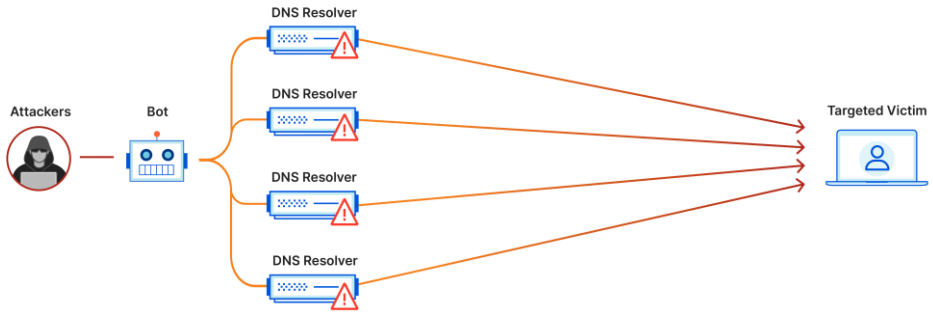
Outra medida essencial é a configuração de firewalls e sistemas de prevenção de intrusões (IPS). Ferramentas como Web Application Firewalls (WAFs) podem identificar e bloquear tráfego suspeito, mitigando ataques antes que eles causem danos significativos. Tanenbaum e Wetherall (2011) ressaltam que firewalls configurados corretamente ajudam a limitar conexões maliciosas, protegendo a integridade do sistema. A monitorização contínua do tráfego de rede também é crucial para a detecção precoce de padrões anômalos que possam indicar um ataque em andamento. Ferramentas como Wireshark, Splunk e Zabbix permitem que administradores analisem o tráfego e identifiquem aumentos repentinos ou picos incomuns. Com base nesses alertas, respostas automáticas ou manuais podem ser ativadas para mitigar o ataque (Whitman & Mattord, 2018).

Manter softwares atualizados e remover programas desnecessários são práticas fundamentais para reduzir a superfície de ataque e evitar a exploração de vulnerabilidades. A CERT.BR (2022) destaca que a desativação de serviços não essenciais e o uso de patches de segurança atualizados são passos importantes para

proteger a infraestrutura contra ataques. A criação de planos de resposta a incidentes é indispensável para garantir que a organização esteja preparada para lidar com ataques DoS e DDoS. Esses planos devem incluir estratégias como o reroteamento de tráfego, ativação de servidores de backup e a comunicação com o provedor de serviços de internet para suporte adicional durante o ataque. Tanenbaum e Wetherall (2011) reforçam que respostas rápidas e bem planejadas são fundamentais para minimizar danos.

Por fim, a distribuição da infraestrutura e o uso de redes de entrega de conteúdo (CDNs) podem aumentar significativamente a resiliência contra ataques volumétricos. A distribuição geográfica dos servidores ajuda a dispersar o tráfego malicioso, reduzindo o impacto do ataque. Soluções de escalonamento automático em ambientes de nuvem também são eficazes para lidar com picos de tráfego, sejam legítimos ou maliciosos (Stallings, 2017).

Figura 3 - Exemplo de ataque DDoS



Fonte: CLOUDFLARE (2023).

A Figura 3 representa um ataque de negação de serviço distribuído (DDoS), que é um tipo de ataque cibernético. O processo é o seguinte:

- **Hacker -** Representado no topo, este é o indivíduo ou grupo que coordena o ataque. Eles usam um computador para controlar outros computadores remotamente.
- **Computadores Mestres -** São computadores infectados pelo hacker e usados para controlar ainda mais máquinas. Esses computadores distribuem as ordens do hacker para os computadores subordinados.

- **Computadores Infectados -** Estes são os computadores que foram infectados por malware sem o conhecimento de seus proprietários. Eles recebem comandos dos computadores mestres para atacar o alvo.
- **Alvo -** Este é o computador ou servidor que é o destino final do ataque. Os computadores infectados enviam solicitações simultâneas para sobrecarregar o sistema alvo, fazendo com que ele fique indisponível para usuários legítimos.

2.5. Códigos maliciosos (Malware)

Malware é um termo genérico que engloba todo tipo de software malicioso projetado para danificar dispositivos, sequestrar dados, bombardear com anúncios, interceptar informações ou simplesmente irritar (CERT.BR, 2012). Existem vários tipos de malwares, como vírus, cavalo de Troia (trojan), backdoor, spyware, worm, bot, botnet, ransomware e rootkit, cada um com características próprias que os definem e os diferenciam dos outros (CERT.BR, 2012).

As ações mais comuns da infecção por malwares são a execução automática de mídias removíveis, como pendrives, o direcionamento a sites falsos ou infectados e a execução de arquivos previamente infectados (CERT.BR, 2012). Para se proteger contra malwares, é recomendável manter os programas antivírus e antispyware atualizados, remover programas desnecessários, utilizar apenas programas originais, criar um disco de emergência, verificar periodicamente os logs gerados pelo firewall pessoal, sistema operacional e antimalware, ser cuidadoso ao clicar em links, independentemente de como foram recebidos e de quem os enviou, e desabilitar a auto execução de arquivos anexados e a execução automática de mídias removíveis (CERT.BR, 2012).

Além disso, é importante manter os sistemas e aplicativos atualizados, pois aplicar correções de segurança pode evitar que os dispositivos sejam infectados (CERT.BR, 2020). Em caso de suspeitas de problemas, é recomendável usar o antivírus instalado em seu dispositivo ou opções online imediatamente.

2.5.1. Vírus

Os vírus cibernéticos são programas maliciosos que se infiltram em

computadores e dispositivos móveis sem o consentimento do usuário, causando diversos tipos de danos, como a exclusão ou alteração de arquivos, roubo de informações confidenciais e a inutilização do dispositivo. De acordo com a Cartilha de Segurança para Internet do CERT.br, os vírus cibernéticos são uma ameaça significativa à segurança das redes e sistemas de informação, podendo causar prejuízos financeiros e danos à reputação das organizações afetadas (CERT.BR, 2022)

A disseminação de vírus cibernéticos pode ocorrer por meio de diversos vetores, como e-mails maliciosos, downloads de arquivos infectados, acesso a sites comprometidos ou mesmo por meio de dispositivos de armazenamento externo contaminados. Além disso, a evolução das técnicas de ataque e a sofisticação dos códigos maliciosos tornam cada vez mais difícil a detecção e a remoção de vírus cibernéticos (CERT.BR, 2022).

Para se proteger contra vírus cibernéticos, é necessário adotar medidas preventivas, como a instalação e atualização regular de softwares de segurança, a conscientização dos usuários sobre práticas seguras na internet e a adoção de políticas de segurança que garantam a proteção dos sistemas e dados (Prodemge, 2017). Além disso, a educação dos usuários sobre as técnicas de ataque mais comuns e a importância de manter os sistemas e softwares atualizados é fundamental para a prevenção de infecções por vírus cibernéticos (Prodemge, 2017).

2.5.2. Worms

Worms são um tipo de malware que pode se replicar e se espalhar em redes sem a necessidade de intervenção humana. Eles podem causar danos significativos aos sistemas e redes de computadores, como perda de dados, travamentos de sistemas e congestionamento de rede. Worms podem explorar vulnerabilidades em sistemas operacionais, aplicativos de software e protocolos de rede para ganhar acesso a sistemas e se espalhar (CERT.BR, 2022).

De acordo com o Comitê Gestor da Internet no Brasil (CGI.br), worms são uma ameaça significativa à segurança digital e seu impacto pode ser grande, especialmente em organizações com redes grandes. Para mitigar o risco de worms, é essencial implementar medidas robustas de segurança digital, incluindo corta-fogos, sistemas de detecção de intrusão e software antivírus (CERT.BR, 2022).

Worms também podem ser espalhados por meios de engenharia social, como emails de phishing ou mensagens instantâneas, que enganam usuários para baixar ou executar o malware. Portanto, a educação e a conscientização dos usuários são fundamentais para prevenir infecções por worms. Os usuários devem ser treinados para reconhecer e evitar emails, links e downloads suspeitos e manter seus softwares e sistemas atualizados com as últimas patches de segurança (CETIC.BR, 2020).

Além das medidas técnicas e da educação do usuário, as organizações também devem ter planos de resposta a incidentes em vigor para detectar e responder rapidamente a surtos de worms. Esses planos devem incluir procedimentos para identificar a fonte do worm, conter sua propagação e remediar quaisquer danos causados (CETIC.BR, 2020).

2.6. Gamificação

A gamificação, termo derivado do inglês "gamification", refere-se à aplicação de elementos de jogos em contextos não relacionados a jogos, como a educação, negócios e saúde. Segundo Kiryakova, Angelova e Yordanova (2014), trata-se de uma estratégia inovadora que visa aumentar a motivação e o engajamento por meio do uso de mecânicas de jogos, como pontuações, níveis, recompensas e desafios. Essa abordagem baseia-se na premissa de que os jogos promovem um engajamento intrínseco devido à sua natureza envolvente e interativa. No contexto educacional, a gamificação oferece uma oportunidade de transformar o aprendizado em uma experiência mais dinâmica e significativa, conectando-se às necessidades de alunos contemporâneos, frequentemente caracterizados como "nativos digitais" (Kiryakova et al., 2014).

Na educação, a gamificação tem sido amplamente explorada como uma ferramenta para melhorar o envolvimento dos alunos e facilitar a aprendizagem ativa. Estudos como os de Langendahl, Cook e Mark-Herbert (2016) destacam que elementos como narrativas, desafios, progressão e feedback são particularmente eficazes em manter a motivação dos alunos. Esses elementos podem ser incorporados em atividades pedagógicas para estimular o pensamento crítico, a resolução de problemas e o aprendizado colaborativo. Além disso, a gamificação permite adaptar o conteúdo às preferências dos alunos, tornando o processo educacional mais inclusivo e alinhado às demandas de uma sociedade tecnológica (Langendahl et al., 2016).

Embora os benefícios da gamificação sejam amplamente reconhecidos, sua implementação enfrenta desafios significativos. Como destacado por Dichev e Dicheva (2017), a falta de compreensão sobre como adaptar efetivamente os elementos de jogos ao contexto educacional pode limitar os resultados. Além disso, a gamificação, quando mal projetada, pode levar a um foco excessivo em recompensas extrínsecas, prejudicando a motivação intrínseca dos alunos. Outro ponto crítico é a necessidade de infraestrutura tecnológica e capacitação docente para o desenvolvimento e gerenciamento de atividades gamificadas. Assim, para maximizar os benefícios, é essencial um planejamento cuidadoso que leve em consideração as especificidades do contexto educacional (Dichev & Dicheva, 2017).

A gamificação apresenta-se como uma estratégia altamente eficaz para enfrentar desafios de engajamento, motivação e retenção de conhecimento em diferentes contextos educacionais e profissionais. Sua capacidade de transformar atividades tradicionais em experiências interativas e significativas justifica seu uso no desenvolvimento de jogos. Como destacado por Paľová e Vejačka (2022), a integração de elementos de jogos no aprendizado cria um ambiente dinâmico e envolvente, alinhado às expectativas de uma geração acostumada a tecnologias digitais. Além disso, os elementos gamificados, como recompensas, desafios e progressão, promovem não apenas a diversão, mas também a aquisição de habilidades e competências específicas (Paľová & Vejačka, 2022). Portanto, ao incorporar a gamificação no desenvolvimento de jogos, é possível criar produtos que transcendem o entretenimento, contribuindo para o aprendizado, a formação de competências e o engajamento do público-alvo. Essa abordagem, fundamentada em evidências e amplamente estudada, reforça a justificativa para o uso dessa técnica como base para o projeto, assegurando sua relevância e eficácia nos objetivos propostos.

3. MATERIAIS E MÉTODOS

Este capítulo descreve os materiais e métodos utilizados no desenvolvimento do jogo "Palavras Cruzadas". A metodologia abrange desde o ambiente de desenvolvimento e as ferramentas empregadas até o processo de criação e validação do jogo, com o objetivo de proporcionar um aprendizado interativo sobre segurança digital.

3.1. Ambiente de Desenvolvimento

O desenvolvimento do jogo foi realizado em uma máquina virtual (VM) com sistema operacional Linux, configurada com 12GB de memória RAM. Essa VM proporcionou um ambiente seguro e isolado para o desenvolvimento, garantindo a estabilidade necessária durante a implementação do projeto. A escolha pelo Linux se deu devido à sua flexibilidade e à disponibilidade de ferramentas de desenvolvimento gratuitas e de código aberto, além da possibilidade de personalização do ambiente.

O editor de código utilizado foi o Visual Studio Code (VSCode), versão 1.48.1. O VSCode oferece suporte a várias linguagens de programação e possui uma vasta biblioteca de extensões que facilitam o desenvolvimento, como depuradores e suporte para Python. Além disso, a versão utilizada do Python foi a 3.10.12, que inclui diversas melhorias de desempenho e novas funcionalidades que facilitaram a construção do projeto.

3.2. Ferramentas Utilizadas

A escolha da linguagem de programação Python (versão 3.10.12) para o desenvolvimento do jogo se deu pela sua simplicidade, flexibilidade e vasta comunidade de suporte. Amplamente utilizada em projetos educativos, o Python possui uma sintaxe amigável, o que facilita tanto o aprendizado quanto a implementação de soluções. Para criar a interface gráfica e gerenciar os elementos do jogo, como eventos, imagens e sons, foi utilizada a biblioteca Pygame (versão 2.6.0), que proporcionou uma experiência de usuário interativa e responsiva. Além disso, para empacotar o jogo como um executável e permitir sua distribuição sem a

necessidade de instalar o Python e suas dependências, foi utilizado o PyInstaller.

Para o gerenciamento do código-fonte, foi adotado o sistema de controle de versão Git, que facilitou a rastreabilidade de alterações e o trabalho colaborativo durante o desenvolvimento. Essa escolha garantiu maior organização e eficiência no processo de criação do jogo, possibilitando um acompanhamento contínuo das modificações e a integração de contribuições de diferentes membros da equipe.

3.3. Metodologia de Desenvolvimento

A metodologia adotada para o desenvolvimento do jogo foi estruturada em várias etapas, começando pelo planejamento inicial, que envolveu a definição dos objetivos do jogo e a escolha da temática de segurança digital como foco principal. Nessa fase, também foram elaborados os conceitos de jogabilidade, o público-alvo e os elementos educacionais que seriam incorporados à experiência. Em seguida, o desenvolvimento do código foi realizado de forma modular, dividindo-o em três principais arquivos: main.py, fase_facil.py e dicas.py. Essa estrutura modular facilitou a organização do trabalho e a manutenção do código, com o arquivo main.py gerenciando o menu principal, o fase_facil.py implementando a fase de jogabilidade e o dicas.py apresentando informações educativas sobre ataques cibernéticos.

Durante o desenvolvimento, o jogo passou por diversas fases de testes e validação para garantir o seu funcionamento adequado. Testes técnicos foram realizados, mas um grupo de 10 usuários, incluindo três com baixo conhecimento sobre segurança digital, também testou o jogo, fornecendo feedbacks importantes sobre a usabilidade e o conteúdo. Com base nesses resultados, foram implementadas melhorias no conteúdo das dicas e na interface do jogo, tornando-os mais acessíveis e compreensíveis para todos os tipos de jogadores. Ajustes específicos foram feitos para tornar algumas dicas mais objetivas e para garantir uma navegação intuitiva pelas funcionalidades e pelo menu do jogo, assegurando a experiência de usuário de forma eficaz.

3.4. Procedimentos Adotados

Durante o desenvolvimento do projeto, foram adotados procedimentos

específicos para garantir tanto a qualidade do código quanto a eficácia dos objetivos educacionais do jogo. A divisão do código em módulos (main.py, fase_facil.py, dicas.py) permitiu que cada parte do jogo fosse desenvolvida e testada separadamente, o que facilitou a manutenção e a depuração. Para assegurar a qualidade do conteúdo educacional, as dicas sobre segurança digital foram elaboradas com base em pesquisas sobre os principais tipos de ataques cibernéticos e as medidas de proteção mais eficazes. A linguagem foi simplificada para tornar as informações acessíveis, principalmente para o público sem conhecimentos prévios sobre segurança digital.

Além disso, o uso do sistema de controle de versão Git permitiu gerenciar as versões do código e documentar o histórico de alterações, facilitando a colaboração e a identificação de problemas durante o desenvolvimento. Esses procedimentos garantiram que o processo de criação do jogo fosse organizado e eficiente, com foco na construção de uma experiência de aprendizado interativa e informativa. O conjunto de métodos e materiais adotados foi essencial para a validação e o sucesso do produto educativo, alinhado aos objetivos estabelecidos inicialmente.

4. DESENVOLVIMENTO

Neste capítulo, será descrito o processo de desenvolvimento do jogo "Palavras Cruzadas", que foi implementado utilizando a biblioteca Pygame em Python. O objetivo do jogo é educar os jogadores sobre segurança da informação por meio de uma atividade interativa. Ao longo deste texto, serão explicadas as etapas de criação, a implementação das principais funcionalidades, como a tela de menu, a fase fácil do jogo, e o gerenciamento das interações do usuário. Serão detalhadas as funções responsáveis pela renderização da interface gráfica, manipulação de eventos e controle do estado do jogo, além de uma explicação sobre a lógica da grade de palavras cruzadas e a validação das respostas do jogador. Ao final, será apresentado o fluxo completo do jogo, com ênfase no funcionamento das diferentes partes que compõem a aplicação.

4.1. Estrutura do Código

A estrutura do código do jogo "Palavras Cruzadas" é dividida em três principais arquivos o primeiro é o main.py, responsável por inicializar o jogo e gerenciar o menu principal, fase_facil.py, que implementa a fase de jogabilidade, e dicas.py, que fornece as informações relacionadas aos ataques cibernéticos. No main.py, são realizadas a configuração da janela, o carregamento de recursos e a interação com o usuário por meio de eventos do mouse e teclado. O arquivo fase_facil.py lida com a lógica do jogo em si, incluindo a criação da grade de palavras cruzadas, a validação das palavras e a renderização dos elementos gráficos. O dicas.py é responsável por exibir as dicas sobre segurança digital, incluindo explicações sobre os diferentes tipos de ataques e como se proteger. Todos os arquivos utilizam funções auxiliares para modularizar o código e facilitar a manutenção, garantindo um desenvolvimento organizado e a possibilidade de futuras expansões.

4.2. Arquivo Main.py

O arquivo *main.py* desempenha um papel crucial na inicialização do jogo e na apresentação do menu principal ao jogador. No menu inicial, o jogador pode escolher

entre iniciar a fase fácil do jogo ou visualizar as dicas. O código implementa essas funcionalidades de forma modular e organizada, facilitando a manutenção e a expansão do jogo.

A execução do código começa com a importação de módulos essenciais para o funcionamento do jogo, incluindo a biblioteca Pygame, que oferece suporte a gráficos, sons e interações, o módulo *sys*, que permite acesso a funções do interpretador Python, como o *sys.exit()* para encerrar o programa, e a função *fase_facil_game*, importada do módulo *fase_facil*, que inicializa a fase fácil do jogo. O módulo *os* também é utilizado para manipulação de caminhos de arquivos, garantindo que recursos, como imagens e sons, sejam encontrados corretamente.

A configuração do caminho dos recursos é feita utilizando *getattr(sys, '_MEIPASS', os.path.dirname(os.path.abspath(__file__)))*, garantindo que os arquivos sejam corretamente referenciados, mesmo quando o programa é empacotado como executável por ferramentas como PyInstaller como foi demonstrado na figura 4 .Esse mecanismo assegura que o programa encontre os arquivos de imagem, áudio e outros recursos, independentemente do ambiente de execução. O Pygame é inicializado com *pygame.init(),* preparando os módulos necessários para a execução do jogo. A janela de exibição é configurada com as dimensões de 1500x800 e definida como redimensionável por meio da função *pygame.display.set_mode((SCREEN_WIDTH, SCREEN_HEIGHT), pygame.RESIZABLE)*. Além disso, o título da janela é configurado como "Palavras Cruzadas - Tela de Início", criando uma interface amigável para o jogador logo na tela de abertura.

Figura 4 - Exemplo de configuração de recursos e janela no Pygame

```
1  import pygame
2  import sys
3  from fase_facil import main_game as fase_facil_game
4  from dicas import mostrar_dicas  # Importa a função para exibir dicas
5  import os
6
7  # Caminho relativo ao diretório onde o executável está rodando
8  base_path = getattr(sys, '_MEIPASS', os.path.dirname(os.path.abspath(__file__)))
9  background_path = os.path.join(base_path, 'background.png')
```

Fonte: Autor.

A imagem de fundo do menu principal é carregada usando *pygame.image.load(background_path)*, e as fontes de texto são configuradas para

melhorar a experiência visual do jogador. Para os títulos, é utilizada uma fonte maior, configurada com *pygame.font.Font(None, 74)*, enquanto os textos menores, como os botões e informações, utilizam uma fonte de tamanho 50, configurada com *pygame.font.Font(None, 50)*. A renderização de texto na tela é facilitada pela função *draw_text(text, font, color, surface, x, y)*, que centraliza o texto na superfície do jogo. Essa função utiliza *font.render(text, True, color)* para renderizar o texto e o posiciona na tela com precisão usando *surface.blit(textobj, textrect)*, o que garante uma apresentação visual clara e profissional para o usuário, tudo isso pode ser visualizado na figura 5.

Figura 5 - Exemplo de carregamento de imagem e configuração de fontes no Pygame

```
11  # Inicializa o Pygame
12  pygame.init()
13
14  # Dimensões da tela
15  SCREEN_WIDTH = 1500
16  SCREEN_HEIGHT = 800
17
18  # Configuração da tela
19  screen = pygame.display.set_mode((SCREEN_WIDTH, SCREEN_HEIGHT), pygame.RESIZABLE)
20  pygame.display.set_caption("Palavras Cruzadas - Tela de Início")
21
22  # Carrega a imagem de fundo
23  background = pygame.image.load(background_path)
24
25  # Fonte
26  font = pygame.font.Font(None, 74)
27  small_font = pygame.font.Font(None, 50)
28
29  # Função para desenhar texto
30  def draw_text(text, font, color, surface, x, y):
31      textobj = font.render(text, True, color)
32      textrect = textobj.get_rect(center=(x, y))
33      surface.blit(textobj, textrect)
```

Fonte: Autor.

4.2.1. Menu Inicial

A função *main_menu()* gerencia o menu principal, mantendo-o em execução até que o jogador decida sair do jogo. O menu é responsivo, permitindo que a interface se ajuste ao redimensionamento da janela. A figura 6 mostra o funcionamento da responsividade da tela do jogo, o tamanho da janela é obtido usando *screen.get_size()* e a imagem de fundo é redimensionada para preencher toda a tela com *pygame.transform.scale(background, (screen_width, screen_height))*. O jogo captura

a posição do mouse com *pygame.mouse.get_pos()* e detecta cliques com *pygame.mouse.get_pressed().*

Figura 6 - Exemplo de responsividade no menu principal do Pygame

```
40      screen_width, screen_height = screen.get_size()
41
42      # Redimensiona a imagem de fundo para caber na tela
43      background_resized = pygame.transform.scale(background, (screen_width, screen_height))
44      screen.blit(background_resized, (0, 0))
45
46      # Desenha os botões
47      mouse = pygame.mouse.get_pos()
48      click = pygame.mouse.get_pressed()
```

Fonte: Autor.

Dois botões são exibidos o de "Jogo" e "Dicas". O botão "Jogo" inicia o jogo chamando *fase_facil_game(screen, draw_text)*, enquanto o botão "Dicas" chama o módulo *dicas.py* para exibir informações sobre ataques cibernéticos. Ambos os botões mudam de cor ao passar o cursor sobre eles, fornecendo feedback visual ao jogador.

4.2.2. Atualização da Tela e Tratamento de Eventos

Após o desenho dos elementos, *pygame.display.update()* é chamado para atualizar a tela. Eventos como *pygame.QUIT* (fechar a janela) e *pygame.VIDEORESIZE* (redimensionar a janela) são tratados adequadamente para encerrar o programa ou ajustar o tamanho da tela conforme necessário. Além disso, o jogo responde a eventos de interação do jogador, como movimentos do mouse e cliques, permitindo uma experiência dinâmica e interativa. Dessa forma, o jogo garante uma interface responsiva e fluida, promovendo uma boa experiência de uso.

4.3. Arquivo Fase_facil.py

Nessa fase, o jogador deve preencher uma grade de palavras cruzadas utilizando os termos corretos relacionados ao tema, com base nas dicas fornecidas ao longo do jogo. A grade é composta por palavras-chave que abrangem aspectos importantes de segurança digital, como "MALWARE" e "FIREWALL", entre outros. O jogador precisa inserir as letras nas posições corretas, respeitando as direções horizontal ou vertical definidas no jogo. O código também inclui funções para verificar se as palavras foram corretamente preenchidas e bloqueia as células já resolvidas,

mantendo o progresso do jogador. Além disso, o arquivo trata da renderização gráfica da grade e gerencia a interação do jogador por meio de eventos capturados pelo Pygame, como digitação, seleção de células e rolagem das dicas, garantindo uma experiência dinâmica e educativa.

4.3.1. Importação e Inicialização

Assim como no Main.py, o Pygame e o módulo sys são importados e o Pygame é inicializado com pygame.init(). Isso garante que todos os módulos estejam prontos para uso.

Várias cores são definidas em formato RGB, como LIGHT_GRAY, DARK_GRAY, WHITE, e BLACK, que são usadas para desenhar elementos visuais no jogo. A fonte padrão utilizada para o texto é configurada como pygame.font.Font(None, 28).

4.3.2. Configuração da Grade do Jogo

A figura 7 mostra como é a grade de palavras cruzadas ela representada por uma matriz bidimensional, onde cada célula pode conter:

- ' ' para células vazias que o jogador deve preencher.
- Letras como 'S', 'E' para células já preenchidas (exemplo: a palavra "SECURITY" já está inserida).
- Um dicionário words mapeia cada palavra às suas coordenadas e direção na grade (horizontal ou vertical), enquanto outro dicionário clues armazena as dicas relacionadas a essas palavras.

Figura 7 - Configuração da grade de palavras cruzadas do jogo

```python
# Matriz de palavras cruzadas (com 'security' já preenchido e imutável)
initial_grid = [
    [' ',' ',' ', ' ', ' ', ' ', ' ', ' ', ' ', ' ', ' ', ' ', ' ', ' ', ' ', ' ', ' ', ' '],
    [' ',' ',' ', ' ', ' ', ' ', ' ', ' ', ' ', 'S', ' ', ' ', ' ', ' ', ' ', ' ', ' ', ' '],
    [' ',' ',' ', ' ', ' ', ' ', ' ', ' ', ' ', 'E', ' ', ' ', ' ', ' ', ' ', ' ', ' ', ' '],
    [' ',' ',' ', ' ', ' ', ' ', ' ', ' ', ' ', 'C', ' ', ' ', ' ', ' ', ' ', ' ', ' ', ' '],
    [' ',' ',' ', ' ', ' ', ' ', ' ', ' ', ' ', 'U', ' ', ' ', ' ', ' ', ' ', ' ', ' ', ' '],
    [' ',' ',' ', ' ', ' ', ' ', ' ', ' ', ' ', 'R', ' ', ' ', ' ', '-', ' ', ' ', ' ', ' '],
    [' ',' ',' ', ' ', ' ', ' ', ' ', ' ', ' ', 'I', ' ', ' ', ' ', ' ', ' ', ' ', ' ', ' '],
    [' ',' ',' ', ' ', ' ', ' ', ' ', ' ', ' ', 'T', ' ', ' ', ' ', ' ', ' ', ' ', ' ', ' '],
    [' ',' ',' ', ' ', ' ', ' ', ' ', ' ', ' ', 'Y', ' ', ' ', ' ', ' ', ' ', ' ', ' ', ' '],
    [' ',' ',' ', ' ', ' ', ' ', ' ', ' ', ' ', ' ', ' ', ' ', ' ', ' ', ' ', ' ', ' ', ' '],
    [' ',' ',' ', ' ', ' ', ' ', ' ', ' ', ' ', ' ', ' ', ' ', ' ', ' ', ' ', ' ', ' ', ' '],
    [' ',' ',' ', ' ', ' ', ' ', ' ', ' ', ' ', ' ', ' ', ' ', ' ', ' ', ' ', ' ', ' ', ' '],
    [' ',' ',' ', ' ', ' ', ' ', ' ', ' ', ' ', ' ', ' ', ' ', ' ', ' ', ' ', ' ', ' ', ' '],
]

# Palavras e suas posições (palavra, linha, coluna, direção 'H' ou 'V')
words = {
    "MALWARE": (2, 3, 'H'),
    "WORMS": (2, 6, 'V'),
    "DOS": (6, 4, 'H'),
    "DDOS": (6, 4, 'V'),
    "BRUTE-FORCE": (5, 8, 'H'),
    "SNIFFING": (1, 9, 'H'),
    "SECURITY": (1, 9, 'V'),
    "VIRUS": (9, 0, 'H')
}
```

Fonte: Autor.

4.3.3. Lógica do Jogo

A lógica do jogo é gerida por um dicionário denominado game_state, que desempenha um papel crucial na manutenção do estado global da aplicação. Esse dicionário armazena a matriz atual do jogo, ou seja, a grade com as palavras cruzadas, além das palavras já resolvidas pelo jogador e as células bloqueadas, que são aquelas que não podem ser editadas, garantindo a correta evolução da jogabilidade.

Para facilitar a manutenção e modularização do código, diversas funções auxiliares foram implementadas. A figura 8 mostra a função generate_mask(grid, words) é responsável por criar uma máscara que indica quais células da grade estão ativas (devem ser preenchidas pelo jogador) e quais estão inativas (serão desenhadas como células bloqueadas). A função draw_grid(screen, grid, mask, selected_cell, cell_size) lida com o desenho da grade na tela, destacando a célula atualmente selecionada e exibindo corretamente as letras inseridas pelo jogador. Por fim, a função check_word(grid, word, row, col, direction) verifica se uma palavra foi preenchida corretamente na grade, comparando as letras inseridas com as palavras-alvo.

Figura 8 - Funções auxiliares para manipulação da grade de palavras cruzadas

```python
# Gera a matriz que define quais células devem ficar pretas (não serão preenchidas)
def generate_mask(grid, words):
    mask = [[' ' for _ in range(len(grid[0]))] for _ in range(len(grid))]

    for word, (row, col, direction) in words.items():
        for i in range(len(word)):
            if direction == 'H':
                mask[row][col + i] = word[i]
            elif direction == 'V':
                mask[row + i][col] = word[i]

    for row in range(len(grid)):
        for col in range(len(grid[0])):
            if mask[row][col] == ' ' and grid[row][col] != '-':
                mask[row][col] = 'X'  # Marque as células que não serão usadas

    return mask

# Função para desenhar a matriz na tela
def draw_grid(screen, grid, mask, selected_cell, cell_size):
    # Desenha uma borda ao redor da matriz de palavras cruzadas
    pygame.draw.rect(screen, DARK_GRAY, (45, 45, GRID_WIDTH * cell_size + 10, GRID_HEIGHT * cell_size + 10), 3)

    for row in range(len(grid)):
        for col in range(len(grid[0])):
            rect = pygame.Rect(col * cell_size + 50, row * cell_size + 50, cell_size, cell_size)
            if mask[row][col] == 'X':
                pygame.draw.rect(screen, BLACK, rect)  # Preenche as células inativas com preto
            else:
                pygame.draw.rect(screen, BLACK, rect, 2)
                if grid[row][col] != ' ':
                    text = font.render(grid[row][col], True, BLACK)
                    screen.blit(text, (rect.x + cell_size // 4, rect.y + cell_size // 4))
            if selected_cell == (row, col):
                pygame.draw.rect(screen, BLUE, rect, 3)
```

Fonte: Autor.

A função principal do jogo, main_game(screen, draw_text), é responsável por gerenciar o loop principal da fase fácil. Ela atualiza a tela continuamente com a grade de palavras cruzadas e as dicas relacionadas, além de processar todas as entradas do jogador, como a digitação de letras e a navegação pela grade. Durante a execução, essa função também verifica se todas as palavras foram corretamente resolvidas, oferecendo a opção de o jogador retornar ao menu inicial ao concluir a fase.

4.3.4. Tratamento de Eventos

O jogo responde a diversos eventos capturados pela biblioteca Pygame, permitindo uma experiência interativa e fluida. Entre os principais eventos estão os de teclado, com pygame.KEYDOWN capturando as letras digitadas e os comandos de navegação entre as células. O evento de clique do mouse, pygame.MOUSEBUTTONDOWN, permite que o jogador selecione células na grade ou interaja com botões na interface do jogo. Por fim, a função de rolagem do mouse,

pygame.MOUSEWHEEL, controla a rolagem da lista de dicas, permitindo ao jogador navegar pelas pistas disponíveis para resolver as palavras cruzadas.

4.4. Arquivo dicas.py

O arquivo dicas.py é responsável por fornecer informações educativas aos jogadores sobre os principais tipos de ataques cibernéticos e como se proteger contra eles. Quando o jogador seleciona o botão "Dicas" no menu principal, o conteúdo do arquivo dicas.py é exibido, apresentando explicações sobre ameaças como malware, worms, DoS, DDoS, brute-force, sniffing e vírus. Cada tipo de ataque é descrito de maneira clara, com exemplos e orientações práticas sobre as medidas de segurança que podem ser tomadas para minimizar os riscos.

A função principal do arquivo, exibir_dicas(), gerencia a apresentação dessas informações na tela de forma organizada e acessível. O conteúdo é mostrado em formato de texto, utilizando o Pygame para renderizar cada seção. A interface permite ao jogador rolar o texto, caso haja muitas informações, garantindo que ele possa ler todos os detalhes sobre cada ameaça cibernética.

5. RESULTADOS

Após a conclusão do desenvolvimento do jogo "Palavras Cruzadas" utilizando a biblioteca Pygame, foram realizados testes com um grupo de 10 jogadores para avaliar a eficácia da proposta em educar sobre segurança digital de forma lúdica. Os resultados obtidos são apresentados e analisados a seguir, buscando fornecer uma compreensão detalhada do impacto do jogo como ferramenta educativa.

5.1. Resultado do desenvolvimento do jogo

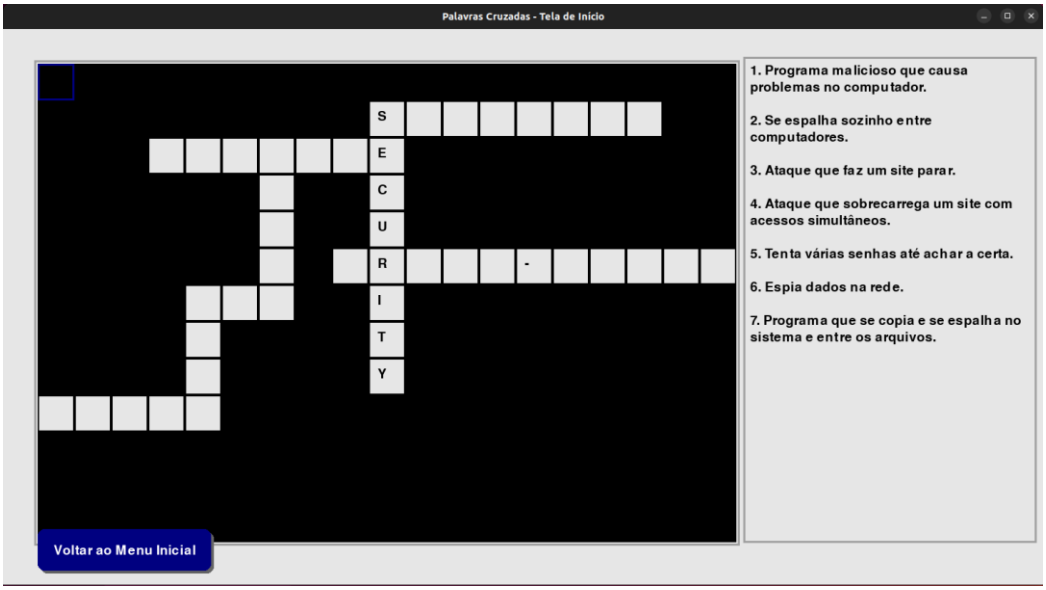
Figura 9 - Tela inicial do jogo

Fonte: Autor.

A tela inicial do jogo "Palavras Cruzadas" representada na figura 9 foi pensada para ser simples e direta. Nela, o jogador encontra duas opções a de "Jogo" e logo abaixo a de "Dicas". A ideia é que o jogador possa escolher entre começar imediatamente a fase principal ou dar uma olhada nas dicas sobre segurança digital antes de jogar.

5.1.1. Tela do Jogo

Figura 10 - Tela principal do jogo

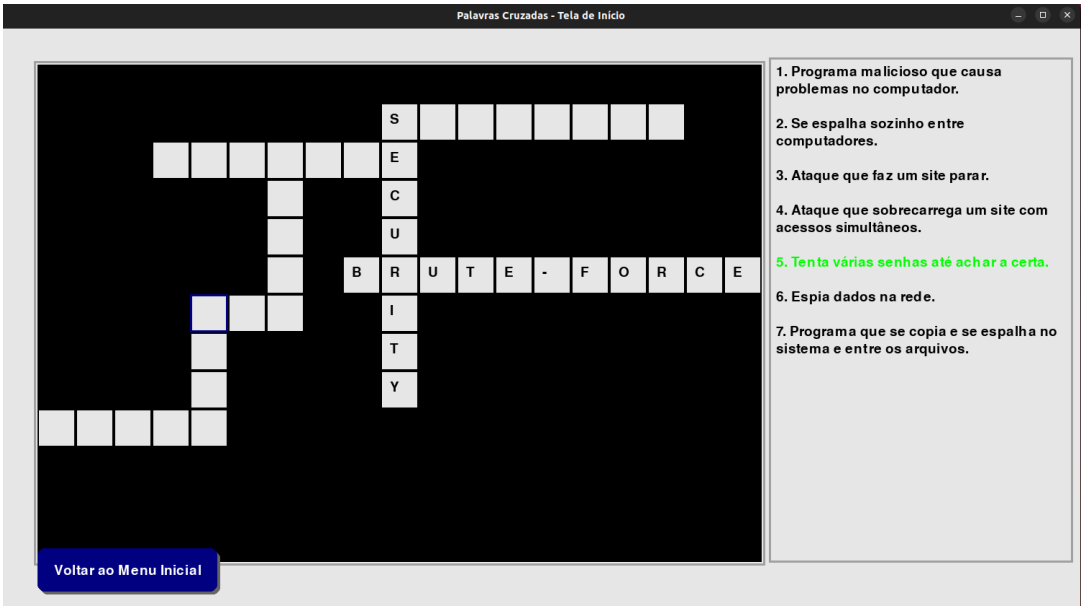


Fonte: Autor.

A Figura 10 apresenta a tela principal do jogo "Palavras Cruzadas". Nessa tela, o jogador encontra uma grade de palavras cruzadas, onde deve preencher os espaços em branco com termos relacionados à segurança digital. À direita da tela, estão as dicas numeradas que guiam o jogador a encontrar as palavras corretas.

O design da tela foi desenvolvido de forma a ser simples e objetivo, garantindo que o foco do jogador permaneça nas palavras cruzadas e nas dicas fornecidas. O botão "Voltar ao Menu Inicial", localizado no canto inferior esquerdo, possibilita que o jogador retorne à tela inicial quando desejar.

Figura 11 - Feedback visual para resposta correta no jogo

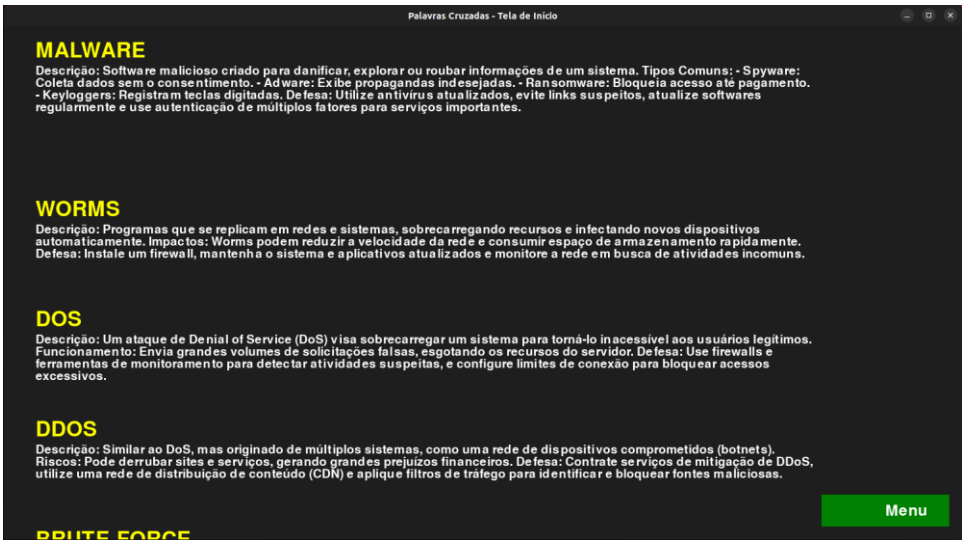


Fonte: Autor.

Quando uma resposta é correta, como ilustrado na figura, a dica correspondente é destacada em verde, proporcionando um feedback positivo imediato ao jogador, como foi mostrado na figura 11.

5.1.2. Tela de Dicas

Figura 12 - Tela de dicas do jogo



Fonte: Autor.

A Figura 12 apresenta a tela de dicas do jogo, que traz informações sobre os principais tipos de ameaças à segurança digital, como malware, worms, DoS, DDoS, e ataques de força bruta. Cada ameaça é descrita de maneira objetiva, juntamente com exemplos e recomendações de defesa.

Essa tela foi criada para oferecer uma maneira simples e clara de aprender sobre segurança digital. O jogador pode explorar as informações e depois usar o botão "Menu" no canto inferior direito para voltar ao menu principal sem complicações.

5.2. Avaliação dos Testes com Usuários

Para avaliar a eficácia do jogo, foram realizados testes com um grupo de 10 pessoas. Durante as sessões de jogo, os participantes tiveram a oportunidade de explorar as funcionalidades e responder às atividades propostas, as quais abordavam temas relacionados à segurança digital, como ataques cibernéticos e práticas seguras de navegação. Todos os jogadores conseguiram concluir o jogo com sucesso, respondendo corretamente a todas as questões, o que resultou em uma taxa de conclusão de 100%.

Dos 10 jogadores, 3 possuíam baixo conhecimento sobre segurança digital, enquanto os outros 7 tinham algum nível de familiaridade com o tema. Apesar das diferenças no conhecimento prévio, todos os participantes conseguiram concluir as atividades e responder corretamente às perguntas. Isso demonstra que o jogo é acessível para pessoas com diferentes níveis de entendimento sobre segurança digital, conseguindo transmitir os conceitos de forma clara e eficiente.

O tempo médio de jogo registrado foi de aproximadamente 30 minutos. Esse valor reflete que o tempo de execução do jogo é adequado, garantindo que os jogadores consigam se engajar nas atividades sem se tornarem desinteressados ou fatigados. O tempo também demonstra que o nível de complexidade das questões e desafios está balanceado, permitindo que os participantes mantenham uma boa fluidez ao longo das fases do jogo.

Além das respostas corretas, foi observado que os jogadores tiveram uma boa compreensão dos conceitos abordados, o que indica que a abordagem gamificada é eficaz para o aprendizado dos temas de segurança digital. Os testes sugerem que o jogo cumpriu seu objetivo de informar os usuários sobre riscos cibernéticos e boas práticas de proteção.

5.3. Feedback dos Jogadores

Para complementar os resultados quantitativos, também foi coletado feedback qualitativo dos jogadores. De maneira geral, os participantes expressaram opiniões positivas sobre o conceito do jogo e sua construção. Eles elogiaram a proposta de combinar o entretenimento de um jogo de palavras cruzadas com conteúdo educativo voltado à segurança digital. Segundo os jogadores, a ideia de utilizar gamificação ajudou a tornar o aprendizado mais leve e atrativo, especialmente para assuntos que costumam ser vistos como técnicos ou complexos.

Os jogadores com baixo conhecimento prévio destacaram que o jogo foi particularmente útil para ajudá-los a entender conceitos que antes eram desconhecidos ou difíceis de compreender. Já os jogadores com mais experiência apontaram que o jogo reforçou conhecimentos já adquiridos, mas de forma divertida e envolvente.

Entretanto, uma crítica relevante mencionada por alguns jogadores foi em relação às dicas fornecidas durante o jogo. Eles consideraram que algumas dicas eram vagas e poderiam ser mais objetivas. Essa observação é importante, pois destaca um ponto que pode ser aprimorado nas futuras versões do jogo. Torna-se essencial revisar e refinar as dicas, tornando-as mais claras e diretas, o que deve ajudar a melhorar a experiência dos jogadores e aumentar a compreensão dos conteúdos abordados.

5.4. Impacto Educacional

Os resultados dos testes indicam que o jogo conseguiu atingir seu principal objetivo educacional que é conscientizar os jogadores sobre a segurança digital. A taxa de acertos de 100% é um indicativo claro de que os jogadores compreenderam os temas apresentados durante a experiência lúdica. A combinação de interatividade com conteúdos educativos se mostrou eficaz para engajar os participantes e garantir que absorvessem informações importantes, como identificar ameaças cibernéticas e adotar boas práticas de segurança.

Outro ponto relevante é o impacto positivo na percepção dos jogadores sobre

os temas de segurança digital. Ao transformar conceitos técnicos em desafios interativos, o jogo conseguiu despertar o interesse dos participantes, fazendo com que eles refletissem sobre a importância de estarem protegidos online. Essa mudança de percepção é essencial para promover um comportamento mais consciente e seguro no ambiente digital.

5.5. Limitações e Melhorias

Apesar dos resultados positivos, foram identificadas algumas limitações durante os testes. O principal ponto levantado foi a falta de objetividade em algumas dicas fornecidas ao longo do jogo. Essa questão dificultou um pouco o progresso dos jogadores em determinados momentos, pois as dicas não ofereciam informações claras o suficiente para facilitar a resolução dos desafios. Como melhoria, será realizada uma revisão cuidadosa das dicas, buscando torná-las mais precisas e adequadas ao contexto de cada desafio.

Outra possível melhoria está relacionada à ampliação do público-alvo. Inicialmente, o jogo foi testado por um grupo reduzido de 10 pessoas, o que fornece apenas uma amostra limitada do impacto da ferramenta. Planeja-se realizar novos testes com um público maior e diversificado, a fim de obter um panorama mais abrangente da eficácia e das possíveis dificuldades encontradas por diferentes perfis de usuários.

6. CONSIDERAÇÕES FINAIS

Este trabalho teve como objetivo desenvolver um jogo educativo utilizando a gamificação para promover a conscientização sobre segurança digital. Através da criação do jogo "Palavras Cruzadas", desenvolvido com a biblioteca Pygame, buscou-se proporcionar uma experiência lúdica e interativa que tornasse o aprendizado sobre ameaças cibernéticas e boas práticas de proteção mais acessível e atrativo. A metodologia envolveu desde a pesquisa sobre os principais tipos de ataques cibernéticos, como malware, phishing, DoS, DDoS, brute-force, sniffing e vírus, até o desenvolvimento do jogo, que utilizou esses conceitos para educar os jogadores de forma prática.

A pesquisa bibliográfica realizada no início do projeto permitiu fundamentar os conceitos apresentados no jogo, e o desenvolvimento prático possibilitou a criação de uma ferramenta educativa que se mostrou eficaz em testes realizados com usuários. O jogo aborda questões como a identificação de golpes, a prevenção de ataques cibernéticos e a adoção de boas práticas de segurança online, utilizando gráficos e desafios interativos para facilitar a assimilação dos conteúdos. Através da gamificação, foi possível promover uma experiência de aprendizado que vai além do formato tradicional, transformando conceitos técnicos em atividades envolventes.

Os testes realizados com um grupo de jogadores mostraram que a proposta conseguiu atingir pessoas com diferentes níveis de conhecimento sobre segurança digital. Mesmo aqueles que não possuíam conhecimento prévio sobre o tema conseguiram assimilar os conceitos apresentados e aplicar as práticas de segurança sugeridas durante o jogo. Esse resultado destaca o potencial da gamificação para incluir indivíduos que muitas vezes se sentem alheios a temas técnicos, contribuindo para torná-los mais preparados para lidar com os riscos no ambiente digital.

Os feedbacks obtidos dos participantes foram, em sua maioria, positivos, ressaltando a eficácia da combinação de aprendizado e entretenimento. No entanto, também foram identificadas algumas críticas construtivas, principalmente relacionadas à clareza das dicas oferecidas ao longo do jogo. Como melhoria futura, planeja-se revisar e tornar essas dicas mais objetivas e claras, facilitando o progresso dos jogadores e aumentando a eficácia da aprendizagem. Além disso, será importante

realizar novos testes com um grupo maior e mais diversificado de participantes, o que permitirá obter um panorama mais abrangente da eficácia do jogo.

O impacto educacional do jogo também merece destaque. Ao transformar conceitos complexos, como ataques cibernéticos e medidas de proteção, em desafios práticos e interativos, foi possível promover uma reflexão significativa sobre a importância da segurança digital. Essa abordagem contribuiu para incentivar um comportamento mais consciente e seguro no uso da internet, algo fundamental nos dias de hoje. A mudança de percepção dos jogadores em relação à segurança online é um indicador de que a proposta tem potencial para contribuir para a formação de uma cultura de segurança digital mais sólida.

As limitações encontradas durante o desenvolvimento e os testes fornecem uma direção clara para os próximos passos. Além da revisão das dicas, é importante considerar a expansão do conteúdo do jogo para cobrir mais tópicos relevantes à segurança digital, bem como a inclusão de novas funcionalidades que possam tornar a experiência ainda mais envolvente. A ampliação do público-alvo e a realização de novos testes também são essenciais para garantir que o jogo seja eficaz para diferentes perfis de usuários, aumentando assim seu alcance e impacto educacional.

Em suma, este trabalho demonstrou que a gamificação, aliada ao desenvolvimento de um jogo educativo, pode ser uma ferramenta poderosa para educar sobre segurança digital. Os resultados obtidos indicam que a proposta é promissora e possui grande potencial de impacto. Espera-se que, no futuro, o projeto possa ser ampliado e adaptado para outras áreas da educação digital, contribuindo para a formação de indivíduos mais preparados e conscientes em relação aos desafios do mundo online.

REFERÊNCIAS BIBLIOGRÁFICAS

AVAST. **O que é um sniffer e como se proteger contra ele?** Disponível em: https://www.avast.com/pt-br/c-sniffer. Acesso em: 25 out. 2024.

CERT.BR. (2012, Versão 4.0). **Cartilha de segurança para internet**. Disponível em: https://cartilha.cert.br. Acesso em: 15 maio. 2024.

CERT.BR. (2020). **CERT.br explica os principais tipos de códigos maliciosos e ensina como proteger dispositivos de ataques.** Disponível em: https://nic.br/noticia/releases/cert-br-explica-os-principais-tipos-de-codigos-maliciosos-e-ensina-como-proteger-dispositivos-de-ataques/. Acesso em: 22 maio. 2024.

CERT.BR. (2022). **Cartilha de Segurança para Internet.** Disponível em: https://cartilha.cert.br/livro/capitulo-7/. Acesso em: 10 junho. 2024.

Cetic.br. **SEGURANÇA DIGITAL**: disponível em: https://cetic.br/media/docs/publicacoes/7/20210514123130/estudos-setoriais-seguranca-digital.pdf. Acesso em: 20 maio. 2024.

Cloudflare. (n.d.). **O que é um ataque de negação de serviço (DoS)?** disponível em: https://www.cloudflare.com/pt-br/learning/ddos/glossary/denial-of-service/. Acesso em: 05 junho. 2024.

CISCO SYSTEMS. **Port Security Configuration Guide**. Disponível em: https://www.cisco.com. Acesso em: 12 dez. 2024.

Cloudflare. **O que é ataque de DDoS?.** Disponível em: https://www.cloudflare.com/pt-br/learning/ddos/what-is-a-ddos-attack/. Acesso em: 25 out. 2024.

DICHEV, Christo; DICHEVA, D. **Gamifying education: what is known, what is**

believed and what remains uncertain: a critical review. International Journal of Educational Technology in Higher Education, v. 14, p. 1-36, 2017. Disponível em: https://consensus.app/papers/gamifying-education-what-is-known-what-is-believed-and-what-dichev-dicheva/d71264056b6e523795e48d8fd49cca3c/?utm_source=chatgpt. Acesso em: 12 dez. 2024.

KIRYAKOVA, G.; ANGELOVA, N.; YORDANOVA, L. **Gamification in education**. 2014. Disponível em: https://consensus.app/papers/gamification-in-education-kiryakova-angelova/281ee96fd83b5179a2344c324b1de34e/?utm_source=chatgpt. Acesso em: 12 dez. 2024.

LANGENDAHL, Per-Anders; COOK, M.; MARK-HERBERT, C. **Gamification in higher education**. 2016. https://consensus.app/papers/gamification-in-higher-education-langendahl-cook/75478e9638725793807a495f31854c7e/?utm_source=chatgpt. Acesso em: 12 dez. 2024.

ManageEngine. (2024, 16 de fevereiro). **Ataque de negação de serviço: o que são ataques DoS e como se proteger.** disponível em: https://blogs.manageengine.com/portugues/2024/02/16/ataque-de-negacao-de-servico-o-que-sao-ataques-dos-e-como-se-proteger.html

Microsoft. (n.d.). **O que é um DDoS? Segurança da Microsoft.** disponível em: https://www.microsoft.com/pt-br/security/business/security-101/what-is-a-ddos-attack. Acesso em: 18 julho. 2024.

Ortega-Barón, J.; González-Cabrera, J.; Machimbarrena, J. M.; Montiel, I. Safety.Net: **A Pilot Study on a Multi-Risk Internet Prevention Program. International Journal of Environmental Research and Public Health, 2021.** Disponível em: https://www.mdpi.com/1660-4601/18/8/4249. Acesso em: 19 maio. 2024.

Prodemge. (2017). **Cibersegurança: educação digital e proteção de dados.** Disponível em: https://www.prodemge.gov.br/images/com_arismartbook/download/22/revista_18.pdf. Acesso em: 02 junho. 2024.

Pušelj, D.; Vuković, M.; Suhanek, M. **Internet Security and Safety Learning via an Innovative Educational Game. American Journal of Environmental Science and Engineering, 2019.** Disponível em: https://www.sciencepublishinggroup.com/article/10.11648/j.ajese.20190304.13. Acesso em: 15 abril 2024.

Pygame Documentation. (2023). **Pygame: Python Game Development.** disponível em: https://www.pygame.org/docs. Acesso em: 10 agosto. 2024.

PAĽOVÁ, Dana; VEJAČKA, M. **Implementation of gamification principles into higher education. European Journal of Educational Research**, 2022. Disponível em: https://consensus.app/papers/implementation-of-gamification-principles-into-higher-paľová-vejačka/d8bdbff97e955443b27c6291da439324/?utm_source=chatgpt. Acesso em: 12 dez. 2024.

Scholefield, S.; Shepherd, L. A. **Gamification Techniques for Raising Cyber Security Awareness. 2019.** Disponível em: https://link.springer.com/chapter/10.1007/978-3-030-22351-9_13. Acesso em: 19 maio. 2024.

Shillair, R.; Cotten, S.; Tsai, H.; Alhabash, S.; LaRose, R.; Rifon, N. J. **Online safety begins with you and me: Convincing Internet users to protect themselves. Comput. Hum. Behav., 2015.** Disponível em: https://www.sciencedirect.com/science/article/abs/pii/S0747563215000606?via%3Dihub. Acesso em: 12 março. 2024.

STALLINGS, William**. Cryptography and network security: Principles and practice. 7. ed**. Boston: Pearson, 2017.

TANENBAUM, Andrew S.; WETHERALL, David J. **Computer networks. 5. ed.** Boston: Pearson, 2011.

THE SSL STORE. **A Brute Force Attack Definition & Look at How Brute Force**

Works. Disponível em: https://www.thesslstore.com/blog/brute-force-attack-definition-how-brute-force-works/. Acesso em: 25 out. 2024.

WHITMAN, Michael E.; MATTORD, Herbert J. ***Principles of information security*. 6. ed.** Boston: Cengage Learning, 2018.

Zscaler. (n.d.). **O que é um ataque de negação de serviço (DoS)?** disponível em: https://www.zscaler.com.br/resources/security-terms-glossary/what-is-a-denial-of-service-attack. Acesso em: 15 setembro. 2024.