UNIVERSITY OF PLYMOUTH

PUSL3190 Computing Individual Project

Project Initiation Document

AI-Powered Phishing Detection System

Supervisor: Mr. Chamara Dissanayake

Name: Ashen Abeysekara
Plymouth Index Number: 10899221
Degree Program: BSc (Hons) Computer Security

Table of Contents

1. Introduction

1.1 Overview of the Project

The AI-Powered Phishing Detection System is a solution to a huge problem in the security field. This combines real-time browser protection and advanced AI to create a comprehensive defense against modern phishing attacks. As reported by the Cybersecurity & Infrastructure Security Agency (CISA), over 90% of cyberattacks begin with phishing and according to the Anti-Phishing Working Group (APWG), the number of phishing sites detected reached over 240,000 in the first quarter of 2021 alone, an increase of 20% compared to the previous year. These alarming statistics underscores the urgency for advanced protective measures that can intelligently detect and respond to phishing threats in real-time. The proposed project, titled "AI-Powered Phishing Detection System," which was originally planned to do URL analysis, image analysis, and text analysis was changed to only aim on providing users with a robust tool in the form of a browser extension that evaluates URLs for phishing potential, assigns a risk score, and directs users to an AI-powered chatbot for detailed analysis and education on detected threats by the suggestions of the supervisor. Because of that this project is now only focused on URL aspect of phishing, but I will continue to develop this project for other aspects as well.

1.2 Inspiration and Anticipated Impact

The inspiration for this project comes from both academic research and the overwhelming number of phishing attempts that users face daily. To address this issue, the proposed project develops a browser extension designed to help the safety of URLs in real-time. By providing a suspicious score on a scale from 0 to 100, users will be immediately informed of potential threats as they navigate the web. If a URL score exceeds 30, an alert will be sent to the user to check with the AI-powered chatbot for a detailed analysis. This solution aims to help users, equipping them with tools to make informed decisions before engaging with potentially malicious websites, thereby enhancing online safety.

2. Business Case

2.1 Business Need

The growth of phishing attacks requires the development of innovative and effective solutions to safeguard internet users. Data from cybersecurity firms shows that phishing remains one of the leading causes of data breaches, with nearly 90% of successful breaches resulting from phishing attempts. For individual users, falling victim to phishing scams can lead to irreversible financial losses, identity theft, and emotional distress. For businesses, the consequences can be even more severe, resulting in significant financial losses, regulatory fines, and reputational damage.

Because of the increasing integration of technology into daily activities ranging from online banking to remote work, the demand for cybersecurity measures has never been more critical. The proposed browser extension will fulfill the urgent need for a proactive, user-centric tool that provides real-time analysis and alerts for potential phishing attacks. By providing immediate feedback, users can make decisions based on the status of the websites they visit.

2.2 Business Objectives

The objectives of implementing the AI-Powered Phishing Detection System align closely with business needs in the field of cybersecurity, particularly in enhancing user protection and awareness.

- **Improving Detection Accuracy:** The system will utilize advanced machine learning algorithms capable of real-time URL analysis, specializing in identifying deceptive phishing attempts.
- **Increasing User Engagement:** By incorporating an AI chatbot for deeper analysis, users will be empowered with instant explanations of potential threats, leading to enhanced security awareness.
- **Enhancing Efficiency:** Achieving detection speeds under one second will give smooth user experience and lessen exposure time to potential phishing threats.

3. Project Objectives

The primary objectives of the phishing detection browser extension project are listed below:

1. **Real-time URL Analysis:**
   Develop a system that performs instant evaluations of URLs as users browse, assessing them against a range of criteria indicative of phishing attempts, such as known malicious domains and suspicious characteristics.

2. **User Alerts:**
   Implement an alert system that notifies users when they attempt to access a URL scoring above a fixed threshold (30), enabling them to take preventive measures.

3. **Integration with AI Chatbot:**
   If the URL score is above 30 and the user can choose to get a deeper analysis with the AI-driven chatbot.

4. **User-Friendly Interface:**
   Design an intuitive and visually appealing user interface that simplifies interaction for users with different levels of technological expertise while ensuring that essential features remain accessible.

4. Literature Review

Phishing attacks have become increasingly advanced, exploiting both user behaviors and technical vulnerabilities to deceive individuals into providing sensitive information. At the start of these attacks there is a critical component known as the Uniform Resource Locator (URL). A significant amount of research has been done on developing methods that focus specifically on detecting phishing through URL analysis, using features and machine learning techniques to differentiate between legitimate and malicious links.

4.1 Overview of URL Characteristics in Phishing Detection

Phishing attacks exploit misleading URLs to lure users into revealing sensitive information. URLs serve as a crucial entry point for phishing detection due to their unique verbal and semantic properties. Research highlights that phishing URLs often differ from legitimate ones in length, domain age, use of special characters, and inclusion of misleading brand names. For example, Banik and Sarma (2018) emphasize that phishing URLs frequently employ subdomains or misspelled brand names to mimic legitimate websites. Similarly, Hutchinson et al. (2018) observed that phishing URLs tend to have excessive parameters and unusual domain extensions, making them recognizable from genuine URLs.

The study by Al-Alyan and Al-Ahmadi (2020) further identifies the role of lexical features in detecting phishing URLs. Features such as the presence of hyphens, IP addresses in the domain, and abnormal path lengths are strong indicators of malicious intent. Additionally, the temporal characteristics of URLs, such as domain age and registration duration, provide significant insights. Phishing domains are often newly registered and have shorter lifespans compared to legitimate domains. Understanding these URL characteristics is foundational for developing strong phishing detection systems.

4.2 Features Used in URL-Based Phishing Detection

Feature extraction is a critical step in phishing detection, as it assists the differentiation between legitimate and malicious URLs. The literature categorizes features into three main groups: lexical, host-based, and content-based features.

1. **Lexical Features:** Lexical features are derived from the structure and syntax of URLs. According to Banik and Sarma (2018), these include the length of the URL, the number of

special characters, and the presence of suspicious keywords. Mahajan and Siddavatam (2018) also highlights the significance of URL entropy and the ratio of digits to letters in identifying phishing URLs.

2. **Host-Based Features:** Host-based features relate to the hosting environment of the URL. These include the domain's age, WHOIS information, and the geographical location of the server. Studies such as those by Hutchinson et al. (2018) and Al-Alyan and Al-Ahmadi (2020) demonstrate that phishing domains are often hosted on free or less secure hosting services. Furthermore, frequent changes in DNS records and the absence of HTTPS certificates are strong indicators of phishing activity.

3. **Content-Based Features:** Content-based features analyze the webpage's HTML and JavaScript content. While these are not directly derived from the URL, they provide supplementary information. For instance, a study published by Sahingoz et al. (2018) found that phishing pages often include obfuscated JavaScript code and iframe redirections. However, content-based analysis is computationally intensive and less suitable for lightweight browser extensions.

4.3 Machine Learning and Deep Learning Approaches

Machine learning (ML) and deep learning (DL) have significantly advanced phishing detection by enabling automated analysis of complex patterns in URLs. Different algorithms cater to varying requirements, balancing accuracy and computational efficiency.

1. **Machine Learning Approaches:** Random Forest (RF) is widely used for lightweight phishing detection. Hutchinson et al. (2018) demonstrated that RF achieves high accuracy by combining decision trees to analyze lexical and host-based features. The algorithm's ability to handle feature interactions and its strength against overfitting make it ideal for browser extensions.

   Support Vector Machines (SVM) have also been effective, particularly in scenarios with limited datasets. Banik and Sarma (2018) employed SVM for URL-based phishing detection, using kernel functions to separate legitimate and malicious URLs in high-dimensional feature spaces. However, SVM's computational overhead limits its applicability in real-time scenarios.

2. **Deep Learning Approaches:** Deep learning models, such as Gradient Boosting Classifiers (GBC) and Convolutional Neural Networks (CNNs), stand out in deeper analysis due to their capacity to learn hierarchical feature representations. Al-Alyan and Al-Ahmadi (2020) implemented a deep learning-based system using GBC, achieving superior accuracy in detecting phishing URLs. GBC's iterative refinement process enables it to capture narrow patterns in URL features, making it suitable for chatbot-based analysis.

   These models analyze sequential data, such as URL character sequences, to identify malicious patterns. However, their computational complexity and training time make them less practical for lightweight applications.

4.4 Limitations and Future Directions

Despite significant advancements, phishing detection systems face several challenges. One major limitation is the evolving nature of phishing techniques. Attackers constantly come up with new methods to bypass detection, such as using homographic domains or encrypting phishing URLs. Al-Alyan and Al-Ahmadi (2020) discuss how adversarial techniques can exploit weaknesses in feature extraction, making it harder for models to maintain accuracy over time. Hutchinson et al. (2018) also emphasizes the need for adaptive systems to counter these evolving threats.

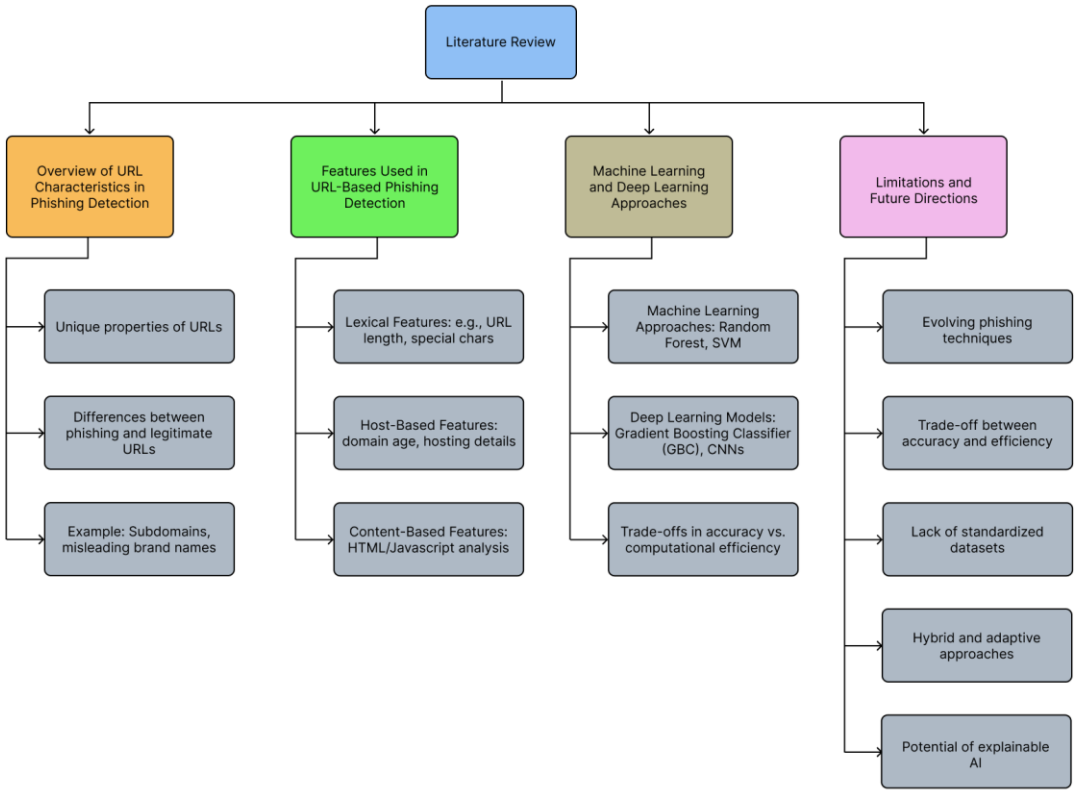
Another challenge is the trade-off between accuracy and computational efficiency. Lightweight models, such as Random Forest, may sacrifice deep analytical capabilities for faster processing, limiting their effectiveness in detecting advanced phishing attempts. Versus, deep learning models like GBC require significant computational resources, making them unsuitable for resource-constrained environments. Sahingoz et al. (2018) suggest that hybrid approaches, combining lightweight models for initial screening and deeper models for detailed analysis, could mitigate this issue.

The lack of standardized datasets is another obstacle. Many studies rely on proprietary datasets, which hinder reproducibility and benchmarking. For example, Banik and Sarma (2018) note the variability in dataset quality and feature availability, which complicates comparisons between different detection systems. Establishing comprehensive, publicly available datasets is essential for advancing phishing detection research.

Future directions include integrating hybrid approaches that combine the strengths of ML and DL models. For example, lightweight models can perform initial screening, while deeper analysis is delegated to advanced models in a distributed framework. Additionally, incorporating real-time threat intelligence feeds can enhance the adaptability of phishing detection systems.

The use of explainable AI (XAI) is another promising way. Providing users with clear explanations for phishing alerts can improve trust and facilitate better decision-making. Moreover, using blockchain technology for secure URL verification could mitigate the risks associated with centralized systems. Banik and Sarma (2018) and Al-Alyan and Al-Ahmadi (2020) emphasize the potential for hybrid and adaptive approaches to improve phishing detection. Additionally, integrating blockchain with AI models has been suggested as a decentralized and tamper-proof solution for phishing detection. Recent studies, such as those by ul Haq et al. (2024) and Nguyen et al. (2014), highlight the importance of combining emerging technologies to address evolving phishing threats. For instance, Banik and Sarma (2018) advocate for URL feature-based detection systems using SVM, while Hutchinson et al. (2018) and ul Haq et al. (2024) suggest the use of random forest and blockchain for robust and decentralized phishing detection systems.

4.5 Conceptual Diagram



This was designed using Figma and it can be viewed from this link if the above image is not clear,

https://www.figma.com/design/mscl8ugaQSf4gDw1SRUsXI/PUSL3190-Lit-review-Conceptual-Diagram?node-id=1-2&t=eKgdTPhAuWiLZxnx-1

5. Method of Approach

5.1 Methodology

The methodology for this project will use the Agile framework, characterized by iterative development and continuous improvement. By dividing the project into sprints, I can adapt to challenges and make necessary adjustments to enhance the end product.

5.2 Technological Framework

The browser extension will be primarily built using JavaScript, using industry standard front-end frameworks like React.js for dynamic user interfaces. The backend services will be developed using FastAPI, which allows for quick and efficient processing of requests, primarily focused on scoring URLs and dispatching alerts.

Machine learning models will support the URL analysis mechanism. These models will analyze various features associated with URLs, such as domain age, length, and similarity to legitimate websites, to classify them as safe or suspicious. Using open-source datasets for training purposes will enable us to develop a strong and effective scoring system.

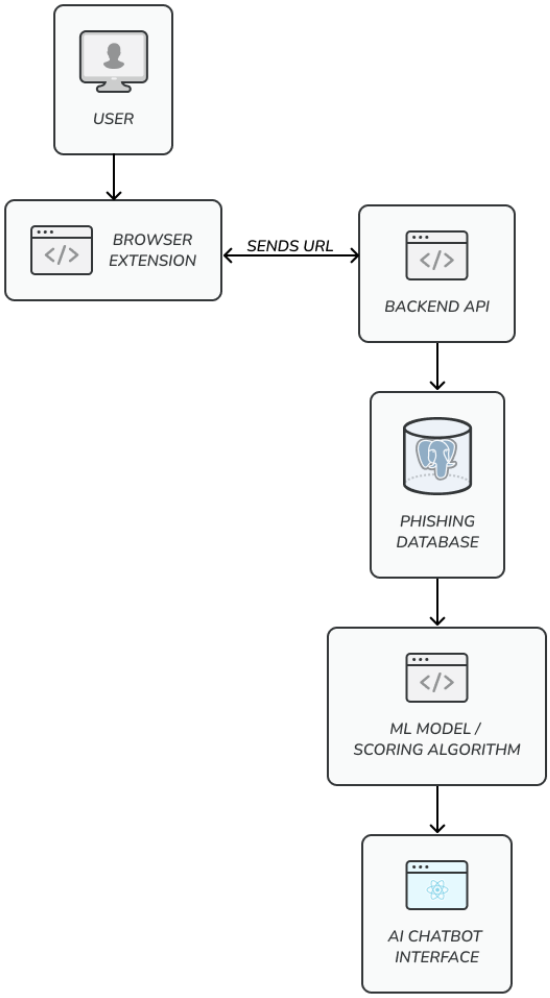
Since these are technological frameworks used in the industry, it will help me a lot to learn about them and power up my skills.

1. **Frontend Technologies**
    - React.js for web interface
    - Chrome Extensions API
    - WebSocket for real-time updates
    - Tailwind CSS for styling

2. **Backend Technologies**
    - Python for AI/ML components
    - FastAPI for REST endpoints
    - Redis for caching
    - PostgreSQL for data storage

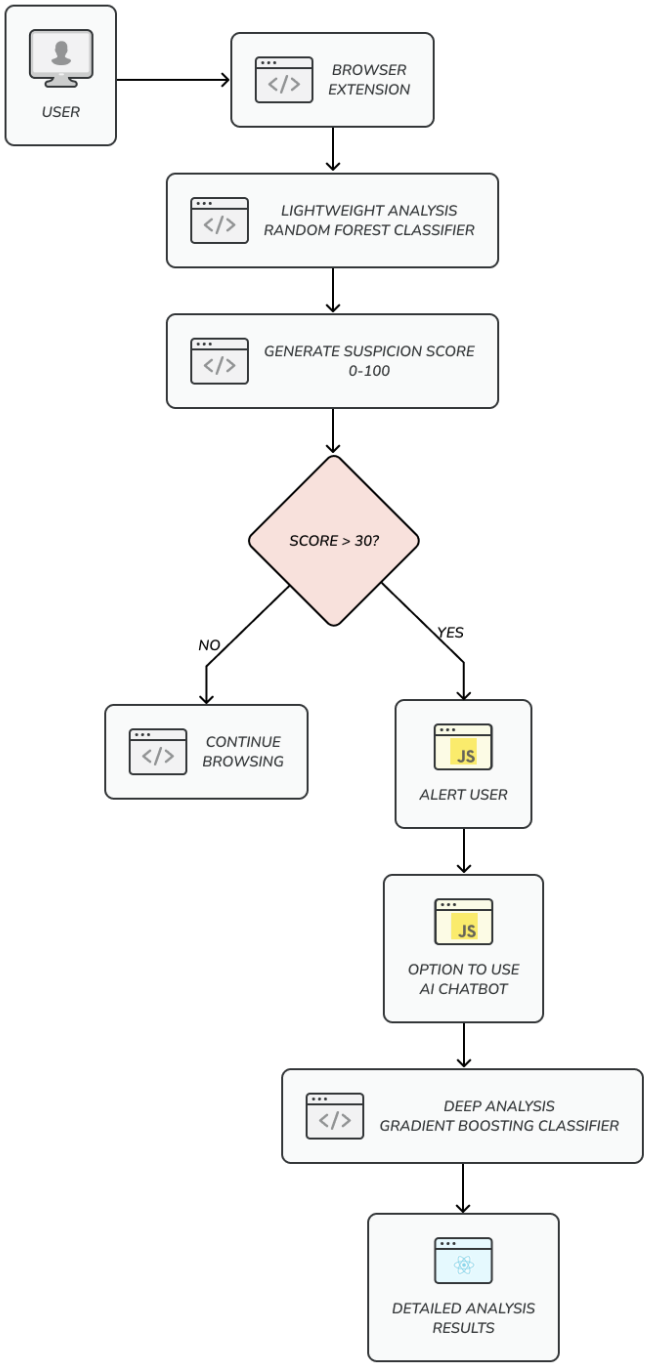
3. **AI/ML Components**
    - Random Forest Classifier for browser extension
    - Gradient Boosting Classifier for deeper analysis in the AI chatbot component.
    - Scikit-learn for traditional ML and implementing random forest classifier & gradient boosting classifier.

5.3 High Level Architectural Diagram



This was designed using Figma and it can be viewed from this link if the above image is not clear, https://www.figma.com/design/QLzEzjz5wBD5R8rEQSFCPp/PUSL3190-High-Level-Architectural-Diagram?node-id=1-2&t=o9yZufqjbDAYpNy7-1

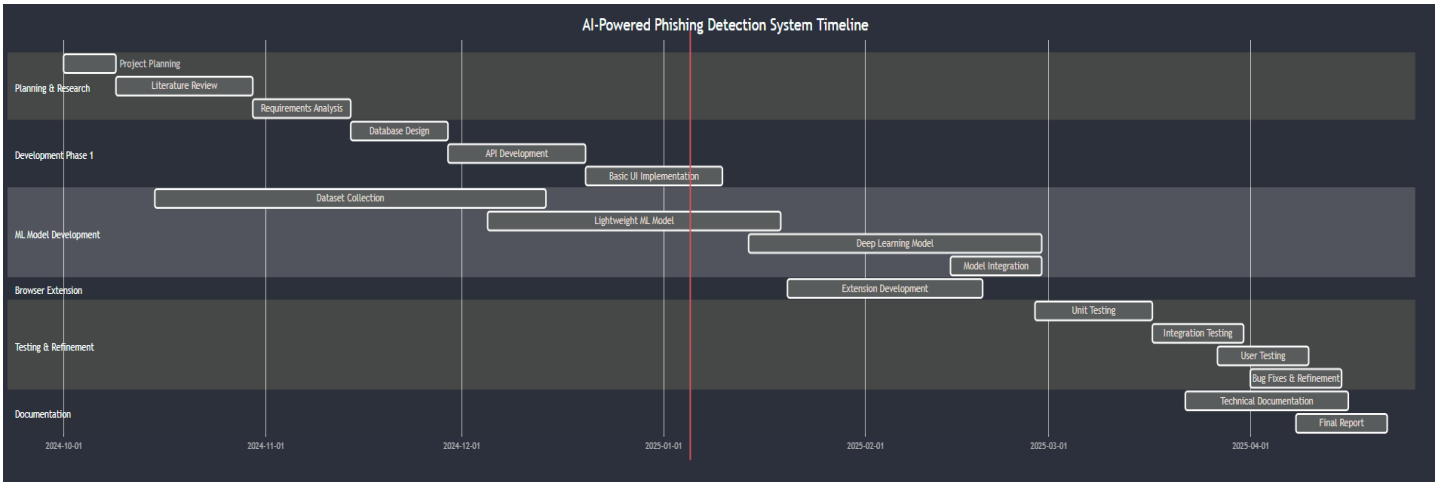
5.4 Process Flow Diagram



This was designed using Figma and it can be viewed from this link if the above image is not clear,

https://www.figma.com/design/cNEEObW8eWyRyQevmDfDkR/PUSL3190-Process-Flow-Diagram?node-id=0-1&t=2ypxWeGtBl0uyRKY-1

6. Initial Project Plan



This Gantt chart was created using mermaid (https://mermaid.js.org). Also made a Gantt chart in excel to track the completion of the project. You can view it using the link below,
AI-Powered Phishing Detection System Gantt chart.xlsx

Details of the time frame if the Gantt chart is not clear,

1. **Planning and Research**
    a. Project planning – 8 days (from October 1st to 9th of 2024)
    b. Literature review – 21 days (from October 9th to 30th of 2024)
    c. Requirements analysis – 15 days (from October 30th to November 14th of 2024)
2. **Development Phase 1**
    a. Database design – 15 days (from November 14th to November 29th of 2024)
    b. API Development – 21 days (from November 29th to December 20th of 2024)
    c. Basic UI implementation – 21 days (from December 20th to January 10th of 2025)
3. **ML Model Development**
    a. Dataset collection - 60 days (from October 15th to December 14th of 2024)
    b. Lightweight ML Model – 45 days (from December 5th to January 19th of 2025)
    c. Deep Learning Model – 45 days (from January 14th to February 28th of 2025)
    d. Model integration – 14 days (from February 14th to February 28th of 2025)

4. **Browser Extension**
    a. Extension Development – 30 days (from January 20th to February 19th of 2025)
5. **Testing and Refinement**
    a. Unit testing – 18 days (from February 27th to March 17th of 2025)
    b. Integration testing – 14 days (from March 17th to March 31st of 2025)
    c. User testing - 14 days (from March 27th to April 10th of 2025)
    d. Bug fixes and refinement – 14 days (from April 1st to April 15th of 2025)
6. **Documentation**
    a. Technical documentation – 25 days (from March 22nd to April 16th of 2025)
    b. Final report – 14 days (from April 8th to April 22nd of 2025)

7. Risk Analysis

Identifying potential risks at the project's beginning allows me to plan and mitigate potential risks. The following risks have been recognized:

1. **Technical Risks**
   - **Likelihood:** Medium
   - **Impact:** High
   - **Description:** Integration of machine learning models may prove complex, particularly in ensuring compatibility between the browser extension and the backend AI components.
   - **Mitigation Strategy:** Regular technical assessments will be conducted throughout the development phases, implementing early identification of integration challenges. Establishing modular design practices allows for easier updates and troubleshooting.

2. **Adoption Risks**
   - **Likelihood:** High
   - **Impact:** Medium
   - **Description:** Users may hesitate to adopt a new phishing detection tool, especially concerning online safety.
   - **Mitigation Strategy:** To build trustworthiness, user engagement initiatives such as tutorials and demo videos will be implemented during the final phase to reassure potential users.

3. **Regulatory Risks**
   - **Likelihood:** Low
   - **Impact:** High
   - **Description:** Non-compliance with data protection laws (e.g., GDPR) can lead to legal repercussions and loss of user trust.
   - **Mitigation Strategy:** The project will prioritize transparent communication about data usage, with design practices adhering to all relevant regulations.

8. Additional Sections

8.1 Stakeholder Analysis

A successful project involves arrangement and engagement with key stakeholders, which include:

- **End Users:** Individuals who will directly use the browser extension to protect themselves from phishing threats. Their feedback will be valuable in enhancing user experience and feature functionality.

- **Organizations and Businesses:** Companies wishing to enhance their cybersecurity protocols will view the extension as a valuable integration, potentially leading to enterprise solutions.

- **Cybersecurity Experts:** Professionals in the field who can provide insight into emerging threats and recommend necessary updates to the extension's algorithms, ensuring that it remains effective against evolving phishing tactics.

8.2 Ethical Considerations

Ethical concerns in the development of the browser extension are paramount and will be systematically addressed as part of the project design. Key considerations include:

- **Privacy and Data Protection:** The project will adhere strictly to data protection regulations, including the General Data Protection Regulation (GDPR) and other relevant laws. The extension will not store or track user behaviors or personal data during URL analyses, focusing solely on real-time interactions.

- **User Transparency:** Clear communication with users about how the extension functions and the nature of its data processing will be essential. Users will be informed about the absence of data collection and the voluntary use of the extension in their browsing activities.

- **User Empowerment:** A core ethical objective is to empower users with knowledge. The project will include educational components that inform users about phishing threats, encouraging safe browsing practices and fostering a proactive cybersecurity mindset.

- **Algorithm Bias:** Efforts will be made to ensure that the machine learning algorithms utilized to assess URLs are trained on diverse datasets, minimizing the risk of bias and ensuring equitable outcomes for all users regardless of demographic factors.

References

- Cybersecurity and Infrastructure Security Agency CISA. (n.d.). Shields Up: Guidance for Families | CISA. [online] Available at: https://www.cisa.gov/shields-guidance families.
- Phishing E-mail Reports and Phishing Site Trends 4 Brand-Domain Pairs Measurement 5 Brands & Legitimate Entities Hijacked by E-mail Phishing Attacks 6 Use of Domain Names for Phishing 7-9 Phishing and Identity Theft in Brazil 10-11 Most Targeted Industry Sectors 12 APWG Phishing Trends Report Contributors 13 Phishing Activity Trends Report Unifying the Global Response To Cybercrime. (2021). Available at: https://docs.apwg.org/reports/apwg_trends_report_q1_2021.pdf.
- Mahajan, R. and Siddavatam, I. (2018). Phishing Website Detection using Machine Learning Algorithms. International Journal of Computer Applications, 181(23), pp.45–47. doi:https://doi.org/10.5120/ijca2018918026.
- Hutchinson, S., Zhang, Z. and Liu, Q., 2018. Detecting phishing websites with random forest. Springer ICST Institute for Computer Sciences, Social Informatics and Telecommunications Engineering MILICOM, 251, pp.470–479.
- Sahingoz, O.K., Buber, E., Demir, O. and Diri, B. (2019). Machine learning based phishing detection from URLs. Expert Systems with Applications, 117, pp.345–357. doi:https://doi.org/10.1016/j.eswa.2018.09.029.
- Al-Alyan, A. and Al-Ahmadi, S., 2020. Robust URL phishing detection based on deep learning. KSII Transactions on Internet and Information Systems, 14(7), pp.2752–2768.
- Banik, B. and Sarma, A., 2018. Phishing URL detection system based on URL features using SVM. International Journal of Electronics and Applied Research (IJEAR), 5(2), pp.40–55. doi:10.33665/IJEAR.2018.v05i02.003.
- Shaik, H.A. (2022). Phishing URL detection using machine learning methods. ResearchGate, [online] p.103288. Available at: https://www.researchgate.net/publication/365790574_Phishing_URL_detection_using_machine_learning_methods [Accessed 3 Nov. 2024].
- Nguyen, L.A.T., To, B.L., Nguyen, H.K. and Nguyen, M.H. (2014). A novel approach for phishing detection using URL-based heuristic. [online] IEEE Xplore. doi:https://doi.org/10.1109/ComManTel.2014.6825621.
- Gupta, B.B., Yadav, K., Razzak, I., Psannis, K., Castiglione, A. and Chang, X. (2021). A novel approach for phishing URLs detection using lexical based machine learning in a real-

time environment. Computer Communications, 175, pp.47–57. doi:https://doi.org/10.1016/j.comcom.2021.04.023.

- Qazi, Faheem, M.H. and Ahmad, I. (2024). Detecting Phishing URLs Based on a Deep Learning Approach to Prevent Cyber-Attacks. Applied Sciences, [online] 14(22), pp.10086–10086. doi:https://doi.org/10.3390/app142210086.