Project Proposal: Phishing Detection Chrome Extension

Project Overview:
A Chrome browser extension designed to enhance online security by detecting and preventing phishing attacks in real-time. The extension monitors URLs, analyzes them for potential threats, and provides users with actionable alerts. By leveraging advanced detection algorithms and real-time analysis, the extension aims to identify suspicious websites, flag phishing attempts, and ultimately protect users from identity theft, data breaches, and other online security risks.

Objectives:

1. **Real-Time URL Monitoring**
   The extension will continuously monitor and analyze the URLs of websites users visit to detect any suspicious patterns associated with phishing attacks (e.g., misleading domain names, suspicious redirects, SSL certificate issues, etc.).
2. **Phishing Detection Algorithms**
   The extension will employ a combination of machine learning models, blacklists, heuristics, and web reputation checks to assess the legitimacy of a website and determine whether it is a phishing attempt.
3. **User Alerts and Notifications**
   Users will receive real-time alerts if they are about to access a potentially harmful site. The alerts will include details such as the nature of the threat, why the website is suspicious, and actions the user can take (e.g., navigate away, report the site).
4. **Reporting System**
   Users can report phishing sites that are not detected by the extension, contributing to the database of threats and helping improve detection accuracy over time.

Key Features:

1. **URL Scanning and Analysis:**
   Continuous scanning of website URLs against a known database of phishing sites and a set of heuristics designed to flag suspicious URLs.
2. **Threat Classification:**
   Classification of threats based on URL patterns, website structure, DNS resolution, and SSL certificate validation.
3. **User Interface (UI):**
   A simple, user-friendly interface that provides:
   - Alerts when phishing is detected.
   - A browser toolbar icon showing the current threat status (safe or phishing).
   - Options to view detailed information about detected threats.
4. **Blacklist and Whitelist Functionality:**
   The ability for users to manually blacklist or whitelist specific websites for personalized protection.
5. **Phishing Education:**
   A section within the extension or popup that educates users about phishing and how to identify potential scams on their own.

Technology Stack:

1. **Frontend:**
   - HTML, CSS, JavaScript for the Chrome extension popup and UI.
   - Chrome APIs for extension development (e.g., `chrome.webRequest`, `chrome.notifications`).
2. **Backend (for URL Analysis and Database):**
   - Integration with threat intelligence APIs (e.g., Google Safe Browsing API, PhishTank, etc.).
   - Optionally, use machine learning models hosted on cloud platforms for real-time phishing detection (TensorFlow, Scikit-learn, etc.).
3. **Real-time Threat Intelligence:**
   Use of threat intelligence sources to continually update the phishing detection algorithms.

Challenges and Solutions:

1. **False Positives/Negatives:**
   - Balancing detection accuracy to minimize false positives and negatives.
   - Regularly updating the detection models and threat databases to adapt to evolving phishing tactics.
2. **User Privacy and Data Security:**
   - Ensuring that the extension does not collect sensitive user data or compromise privacy.
   - Implementing a transparent privacy policy, ensuring that only necessary data (like URL metadata) is collected and anonymized.

Project Milestones:

1. **Phase 1 – Research & Planning:**
   - Research existing phishing detection methods and APIs.
   - Define project requirements and architecture.
   - Create a detailed roadmap for development.
2. **Phase 2 – Extension Development:**
   - Develop the core extension features (URL monitoring, real-time scanning, alerts).
   - Integrate with phishing detection APIs and databases.
3. **Phase 3 – User Interface & UX Design:**
   - Design and implement a clean, user-friendly interface.
   - Provide clear feedback and alerts for users.
4. **Phase 4 – Testing & Optimization:**