DNS Shield Project - AI/ML & SDLC Analysis

1. AI/ML Algorithms Used

For this project, we used a combination of Supervised Machine Learning algorithms, specifically:

- Random Forest: Chosen for its high accuracy in classification tasks, it helps in identifying malicious domains by analyzing various features like domain age, TLD reputation, and historical threat data.

- Logistic Regression: Used for binary classification (safe vs. malicious domains) based on historical threat intelligence.

These algorithms were selected due to their efficiency in handling structured data, interpretability, and ability to detect anomalies in DNS requests.

2. APIs Used

We used mock APIs for domain security checks and real-time threat monitoring. Key APIs include:

1. Domain Reputation API - Checks the entered domain against a global database of known malicious/suspicious domains.
2. Threat Intelligence API - Provides real-time threat updates and domain risk scores.
3. DNS Traffic Monitoring API - Captures and logs DNS requests for anomaly detection.

Usage:
- When a user inputs a domain, the system queries the APIs to retrieve threat scores and classification.
- The APIs feed data into the ML model to update the risk assessment dynamically.
- The threat monitoring API continuously scans DNS requests to identify malicious activities.

3. ML Model Workflow

The ML model follows these steps:

1. Data Collection: The system gathers domain data, including past security reports, WHOIS information, TLD reputation, and phishing patterns.
2. Feature Extraction: Extracts domain-related features (age, registration details, DNS records, previous attack reports, etc.).

3. Training the Model: The supervised ML model is trained on labeled datasets containing safe and malicious domains.
4. Threat Scoring & Classification: When a new domain is inputted, the model calculates its risk score (0-1 scale) based on learned patterns.
5. Decision Making:
   - If the score is low (<0.3), the domain is considered safe.
   - If the score is medium (0.3 - 0.7), further inspection is required.
   - If the score is high (>0.7), the domain is marked as potentially malicious.
6. Alert Generation: If a domain is flagged as high risk, the system triggers an alert and updates the security dashboard.

4. SDLC Model Used

We used the Agile SDLC Model because:

- It allows for continuous iteration and improvements based on real-time threat updates.
- The modular approach helps in adding new security features dynamically.
- Frequent testing ensures high security and reliability.
- The flexibility of Agile helps in integrating new APIs, ML models, and visualization tools as the project evolves.

This SDLC model ensures that our DNS Shield project remains adaptive, secure, and scalable for future enhancements.