SAPD PROJECTS IDEAS

CYBERSECURITY TOPICS - EXAMPLES

A) **Intelligent Firewall**: Design a prototype of an "Intelligent Firewall" that implements the Strategy design pattern to allow the administrator to dynamically swap the network packet analysis logic, moving from a simple filter based on static rules to an "intelligent" analysis that uses a machine learning model to detect anomalies.

B) **Adversarial Gateway**: Implement a "Security Gateway" for an artificial intelligence API that, using the Chain of Responsibility design pattern, forces every incoming request (e.g., an image) to pass through a sequential chain of defense handlers (e.g., a validator, a sanitizer) before it can reach the ML model.

C) **Wi-Fi Sentinel**: Design a "Wi-Fi Sentinel" application (the Observer) that registers with a "Wi-Fi Monitor" (the Subject) and, upon notification of newly detected networks, uses the Strategy pattern to apply interchangeable validation logics in order to identify potential "Evil Twins."

ARTIFICIAL INTELLIGENCE TOPICS - EXAMPLES

A) **Adaptive Security Canvas (Cybersecurity):** Build a platform that observes network traffic and adapts to the context (load, time of day, service criticality), choosing each time how to analyze it. AI can plug in at different points: from simple request enrichment (feature tagging) to anomaly detection or alert prioritization. The design remains open: you can move from static rules to "intelligent" engines without changing the interface. Possible patterns, to choose from: Strategy for engine switching, Chain of Responsibility for inspection steps, Observer for telemetry.

B) **GreenField Advisor (Agricultural sustainability):** Build a system that collects data from field sensors, weather, and images and turns them into operational suggestions to reduce waste (water, energy, fertilizers). AI can estimate needs, identify outliers, or support planning, but remains optional and modular. The architecture should favor model substitutability and compatibility with heterogeneous devices. Possible patterns: Observer for sensor subscriptions, Strategy for interchangeable models, Chain of Responsibility for a data pipeline cleaning → feature engineering → estimation.

C) **AgroChain Bridge (Interoperability in agri-food supply chains):** Build a "bridge" that harmonizes formats and semantics coming from different systems (ERPs, logistics platforms, certifications). AI can help reconcile entities, fill missing data, or suggest mappings, but its insertion is gradual and not mandatory. The focus is to create a flexible canonical model and a stable outward-facing API. Possible patterns: Adapter for connectors, Mediator to coordinate services, Facade for simplified exposure, Chain of Responsibility for staged validations.

D) **CareNet Companion (Support for older adults or people with disabilities):** Build an app that aggregates events from devices and home environments and helps decide when and how to notify caregivers or assistance services. AI can recognize unusual patterns, personalize thresholds, or summarize data into understandable indicators; all while remaining swappable based on privacy and resource constraints. The interface aims for simplicity, leaving room for different levels of intervention (informational, preventive, emergency). Possible patterns: Observer for event streams, State for profiles/habits, Strategy for evaluation engines

E) **EcoFashion Scorecard (Sustainable fashion):** Build a dynamic scorecard that assigns a sustainability score to garments and materials, updating with supply-chain and audit data. AI can estimate impacts when data are incomplete or suggest lower-footprint alternatives, without imposing a single metric. The goal is to make the process transparent and let teams decide weights and criteria. Possible patterns: Decorator to enrich products with badges/indicators, Strategy for scoring schemes, Facade for uniform data access.

F) **DevFlow Assistant (Improving software engineering):** Build an assistant that integrates into the toolchain (VCS/CI/CD) and provides progressive feedback on quality, security, and maintainability. AI can help highlight priorities, suggest refactorings, or summarize large changes, while keeping the option to use traditional tools. The solution should remain extensible to different languages and policies. Possible patterns: Observer for build/PR events, Chain of Responsibility for sequential checks, Template Method for review steps, Command to propose applicable fixes.

G) **EarlyCare Gateway (Early disease diagnosis):** Build a gateway that routes clinical data (text, signals, images) through checks and enrichments before providing decision support. AI can act as an intelligent filter, triage engine, or explainability module, but remains swappable by pathology, device, or domain. Emphasis on privacy, traceability, and configurable response times. Possible patterns: Chain of Responsibility for flow steps, Strategy to choose models, Observer for in-production monitoring, Facade for integration with clinical systems.

BOSS PROJECT

Once you have chosen your preferred group, **you must contact the designated supervisors** to agree on the **specific task assignment** and plan the operational activities.

Each group focuses on a distinct yet interconnected aspect of the overall project, with the common goal of developing a comprehensive computer vision system for autonomous navigation and dynamic adaptation to real-world contexts.

Group 1 – Obstacle Detection (Object Detection / Classification)

Supervisors: Alessio Lorè, Tiziano Albore

Objective:

To design and optimize Computer Vision networks dedicated to the detection and classification of obstacles within images or videos, in order to support autonomous navigation and system safety.

Activities:

- (2.1) Analysis of state-of-the-art architectures for object detection/classification (e.g., YOLO, Faster R-CNN, MobileNet, etc.)
- (2.2) Definition of requirements for obstacle detection: types of obstacles to detect (static, dynamic) and contexts (indoor, outdoor)
- (2.3) Modeling and training of the most suitable networks for the specific task
- Performance optimization for real-time environments or edge devices

Expected Output:

- Robust model for obstacle detection/classification
- Report on precision/recall, inference time, and computational efficiency

Group 2 – Path Detection and Scene Understanding (Scene Understanding / Tracking)

Supervisors: Vera Zizzo, Camilla Zambetti

Objective:

To develop neural networks capable of analyzing the overall scene and identifying the navigable path, taking into account the environmental context and the dynamics of moving objects.

Activities:

- (2.1) Study of architectures for semantic segmentation, pose estimation, and tracking (e.g., DeepLab, HRNet, SORT, etc.)
- (2.2) Analysis of requirements for identifying navigable paths and understanding the context (e.g., interaction between obstacles and traversable spaces)
- (2.3) Modeling networks for dynamic scene understanding and training on real-world datasets
- Integration of video sequences, management of complex environmental contexts, and variability of conditions

Expected Output:

- System capable of mapping safe and traversable paths in both indoor and outdoor environments
- Benchmarking on real video sequences, including variable lighting and complex environments

Group 3 – Model Monitoring and Adaptation (Concept and Data Drift)

Supervisors: Roberta Russo, Bartolo Morgigno

Objective:

To develop solutions for the continuous monitoring of computer vision model performance and their adaptation in the presence of contextual or data variations (drift).

Activities:

- (12.1) Analysis of concept drift: new objects or situations not foreseen in the training data

- Study of automatic drift detection methods in computer vision networks
- Collection and annotation of new real-world data
- (12.2) Study of data drift: changes in observed data (e.g., lighting, seasonal variations)
- Techniques for automatic drift detection (statistical tests, adaptive models
- Development of model update strategies, including:
  - Incremental retraining
  - Automated update pipeline
  - Continuous validation metrics

Expected Output:

- Automated system for model monitoring and adaptation
- Comparative analysis of different drift detection strategies