**Security Champion Cheat Sheet (Java & AI)**

**Formål:** Hurtig verifikation af Pull Requests (PR) før merge til main.

**1. Java & API (SEI CERT & OWASP)**

* **[ ] Input Validering:** Er alle eksterne parametre (fra API eller DB) valideret? Bruges vores **OIO-wrapper-klasser** (f.eks. OIOTextField) i stedet for rå String?
* **[ ] Logging (PII):** Er der risiko for, at personhenførbare data (navn, CPR, adresse) lander i loggen? Tjek om log-statements bruger maskering eller anonymiserede objekter.
* **[ ] SQL & Injection:** Bruges PreparedStatements eller Spring Data JPA? (Ingen string-concatenation i SQL-forespørgsler).
* **[ ] Fejlhåndtering:** Viser API'et stack-traces eller interne systemdetaljer ved fejl? (Brug generiske fejlbeskeder til klienten).

**2. AI & Data (ISO 42001)**

* **[ ] Model Drift/Input:** Hvis koden ændrer på modellens input-features: Er de nye features dokumenteret, og er de saglige (ingen diskriminerende proxy-variable)?
* **[ ] SHAP/LIME:** Hvis modellen er gen-trænet: Er der vedhæftet en ny SHAP-graf? Ser vægtningen af parametre "normal" ud, eller er der pludselige hop i vigtigheden af følsomme parametre?
* **[ ] Metadata:** Er nye datasæt registreret i **Open Metadata** med korrekt klassificering (fx "Fortrolig/Følsom")?

**3. Dependencies & Supply Chain (ISO 27001)**

* **[ ] Nye biblioteker:** Er der tilføjet nye pom.xml afhængigheder? Hvis ja, har **CycloneDX**-scannet markeret dem som sikre (ingen kendte CVE'er)?
* **[ ] Versionering:** Bruges LATEST versioner? (Det er forbudt – brug altid specifikke, fastlåste versionsnumre).

**4. Infrastructure & K8s**

* **[ ] Secrets:** Er der hardcodet passwords, tokens eller API-nøgler i koden eller YAML-filer? (Tjek specifikt for .env eller .yaml ændringer).
* **[ ] Least Privilege:** Har den nye service kun de nødvendige adgangsrettigheder til databasen eller andre microservices?

**Den hurtige "Champion-metode"**

Hvis du er i tvivl under et review, så stil udvikleren disse tre spørgsmål:

1. *"Hvis en angriber kontrollerede det her input, hvad er det værste, der kunne ske?"*
2. *"Hvor gemmer vi de data, som denne funktion håndterer?"*
3. *"Kan vi forklare, hvorfor modellen traf præcis denne beslutning i dette tilfælde?"*