Projet pratique 1

1) Objectif du projet

Vous devez **développer une application simple** dont le but est de **montrer concrètement** ce qui se passe lorsqu'un mécanisme apparemment anodin est utilisé, par exemple :

- accepter des cookies,
- charger un pixel de suivi,
- accepter des *Terms & Conditions* ou une politique de confidentialité.

L'objectif n'est **pas** de protéger l'utilisateur, ni d'optimiser la sécurité.
L'objectif est de **rendre visibles des risques normalement invisibles**, en montrant :

- quelles données sont collectées,
- quand elles le sont,
- par quel mécanisme,
- et quelles conséquences cela peut avoir.

2) Modalités du projet

Le projet doit être réalisé **en équipe de 2 à 4 personnes**. La formation des équipes est laissée à l'initiative des étudiants. Une activité **« choix de groupe »** sera mise en place sur Moodle afin de permettre aux étudiants de s'inscrire dans leur équipe.

La **date de remise** du projet est fixée au **dimanche 1er mars 2026 à 23 h 59**. Tout retard sera pénalisé conformément à la politique décrite dans le plan de cours.

Toute communication sur ce projet se fait avec Samuel Desbiens (s6desbie@uqac.ca, bureau P4-6465).

3) Ce que votre application doit faire (obligatoire)

Votre application doit :

1. **Simuler un comportement réel** (site web, service, extension, application mobile, etc.).
2. **Observer et enregistrer** ce qui se passe quand un utilisateur interagit avec votre projet.
3. **Afficher clairement** les données collectées et les événements déclenchés.

Vous devez être capable de répondre clairement à la question :

> *« Qu'est-ce qui se passe exactement quand une action a lieu ? »*
>
> *Exemple :*
>
> *«Qu'est-ce qui se passe exactement quand je clique sur "Accepter" ?*

Cette action peut donc être :

- *cliquer sur « Accepter » une bannière de cookies,*
- *charger une page contenant un tracking pixel,*
- *ouvrir un courriel,*
- *accepter des conditions d'utilisation,*
- *ou toute autre interaction pertinente.*

Note importante :
Le consentement n'est ***pas toujours demandé****.*
Par exemple, un ***tracking pixel*** *s'active automatiquement, sans clic ni autorisation explicite.*
L'exemple du bouton « Accepter » sert uniquement à illustrer le type de question à laquelle votre projet doit répondre.

Ce qui est attendu de votre projet :

- identifier l'événement déclencheur (clic, chargement, ouverture, etc.),
- montrer les actions invisibles qui en découlent,
- rendre ces actions observables et compréhensibles (logs, données, visualisation).

L'objectif est de transformer une action banale en un enchaînement de faits visibles et explicables.

4) Livrables attendus
Vous devez remettre **3 livrables**.

4.1) Une démonstration fonctionnelle hébergée sur Github :

Vous devez fournir une application qui fonctionne et qui permet de voir :

- une interface utilisateur (site web, page, extension, dashboard),
- des événements enregistrés (logs),
- des preuves visibles de collecte ou de suivi (ex. cookies créés, identifiants générés, requêtes envoyées, données stockées, événements horodatés, etc.).

4.2) Un manuel d'installation et d'utilisation sous format Readme :

Le Readme doit expliquer, sans jargon inutile :

- **La menace étudiée :** Exemple : suivi par cookie, tracking pixel, analyse des T&C.
- **Les données observées :** Quelles données sont collectées ? À quel moment ? Par quel mécanisme ?
- **Les implications :** Qu'est-ce que ces données permettent de faire (Profilage ? Corrélation ? Suivi dans le temps ?) ?
- **Les limites de votre démonstration :** Ce que votre application ne montre pas ou simplifie.
- **Jeu de tests / scénarios reproductibles :** Vous devez fournir **des scénarios clairs** que quelqu'un d'autre peut reproduire.

  Exemple :

  1. Ouvrir la page.
  2. Cliquer sur « Refuser les cookies ».
  3. Observer les données stockées.
  4. Recharger la page.
  5. Cliquer sur « Tout accepter ».
  6. Cliquer sur un bouton « Comparer »
  7. Comparer les scénarios.

  Ces scénarios doivent montrer des différences observables.

4.3) Une Vidéo explicative :

Vous devez fournir une vidéo de maximum 10 minutes qui montre :

- le mécanisme choisi et ce qu'il déclenche ;
- la démonstration en direct (interface + logs + preuves) ;
- ce que vous concluez (risques, implications, limites) ;
- Une correction/atténuation que vous proposez.

La vidéo doit être compréhensible **sans lire le code**.

Notez bien : Moodle n'accepte des vidéos qu'à 50 Mo. Si la vôtre est plus grande, envoyez-nous un lien vers une vidéo Youtube, Panopto ou autre.

5) Contraintes obligatoires (non négociables)

- **Données locales uniquement :**
  Vous ne devez pas stocker les données récoltées sur un serveur distant.
  Toutes les données doivent rester **en local**, par exemple :
  - *localStorage* ou cookies du navigateur,
  - base de données locale (ex. Redis),
  - fichiers locaux (logs).

  Aucune donnée ne doit être envoyée vers un service externe.

- **Le code est le vôtre :**
  Vous ne devez **pas utiliser d'APIs ou de librairies** qui collectent des données à votre place (ex. Hotjar, Google Analytics, Meta Pixel) :

  - les services tiers sont **simulés**,
  - les pixels, cookies ou trackers sont **entièrement contrôlés par vous**,
  - Aucune collecte réelle par un acteur externe n'a lieu.

- **Journalisation explicite :**
  Chaque événement important (création d'un cookie, chargement d'un pixel, acceptation/refus du consentement, lecture d'une politique, etc.) doit être :
  - enregistré (log),
  - horodaté,
  - compréhensible (pas de log obscur ou inutile).

- **Rester légal :**
  - Pas de hacking.
  - Pas de contournement illégal.
  - Pas de solution de protection avancée.

  L'objectif n'est pas de montrer des cas extrêmes ni des attaques sophistiquées.

  Il s'agit de **mettre en évidence les dangers de mécanismes normaux**, banals, auxquels nous sommes habitués et que nous utilisons quotidiennement sans y penser.

6) Évaluation :
Vous serez notés de la manière suivante :

Critère d'évaluation	Description	Points
Compréhension du mécanisme étudié	Identification claire du mécanisme (cookies, tracking pixel, T&C, etc.) et explication correcte de son fonctionnement et de ses enjeux.	**20**
Fonctionnalité du projet	Application fonctionnelle, journalisation explicite et horodatée, données observables et comparables, respect des contraintes (local, simulateur, code propre).	**40**
Qualité du Readme	Identification du problème, proposition de correction réaliste, explication ou démonstration de l'impact, compréhension des compromis.	**15**
Qualité de la vidéo explicative	Vidéo structurée, démonstration claire, explications accessibles sans lecture du code, qualité pédagogique.	**15**
Analyse des risques et implications	Analyse claire des risques, distinction entre données collectées et abus possibles, proposition d'une méthode d'atténuation des risques	**10**
Total		**100**

Règles à suivre :

- **README**
  - Doit être **auto-suffisant** : compréhension complète sans la vidéo.
  - Toute information absente du README est considérée comme non fournie.

- **Code / Application**
  - Doit **fonctionner lors de la correction**.
  - Les logs et preuves doivent être observables sans manipulation complexe.

- **Vidéo**
  - Doit être **auto-suffisante** : compréhension complète sans lire le code.
  - Doit montrer la démonstration réelle, pas uniquement des slides.

Idées de projets pratiques (Pistes de départ)

Les projets ci-dessous sont des **idées de directions possibles**, pas des modèles à reproduire mot pour mot. Chaque équipe doit proposer **sa propre interprétation** et ses propres choix techniques.

Idée 1 — Simulation de consentement aux cookies

Développer un site web simple qui simule une bannière de consentement aux cookies (*Tout accepter / Nécessaires / Refuser*). Le projet montre ce qui change concrètement selon le choix effectué : données stockées, événements déclenchés, identifiants utilisés, différences observables entre les options.

Idée 2 — Démonstration du fonctionnement d'un tracking pixel

Créer une application qui illustre comment un pixel de suivi (ex. image invisible) permet de détecter des ouvertures, des visites ou des interactions. Le projet met en évidence quelles informations peuvent être déduites à partir de mécanismes très simples.

Idée 3 — Analyseur de politiques de confidentialité ou de *Terms & Conditions*

Concevoir un outil (web ou extension) qui analyse automatiquement un texte de politique de confidentialité ou de conditions d'utilisation. L'objectif est d'identifier des éléments potentiellement problématiques ou ambigus et de les rendre compréhensibles pour un utilisateur non expert.

Idée 4 — Visualisation du stockage local (cookies, localStorage)

Développer une application qui permet de visualiser l'évolution des données stockées dans le navigateur au fil des interactions : première visite, consentement, retour sur le site, etc.
Le projet montre comment des données persistent et sont réutilisées dans le temps.

Idée 5 — Parcours utilisateur et corrélation de données

Créer une démonstration qui illustre comment des actions apparemment indépendantes (clics, visites, horaires) peuvent être reliées entre elles pour reconstruire un profil ou un comportement.

Règle commune à toutes les idées

Quel que soit le projet choisi, l'objectif est toujours le même :
rendre visibles et compréhensibles des mécanismes de collecte ou d'exploitation de données qui sont habituellement invisibles pour l'utilisateur.