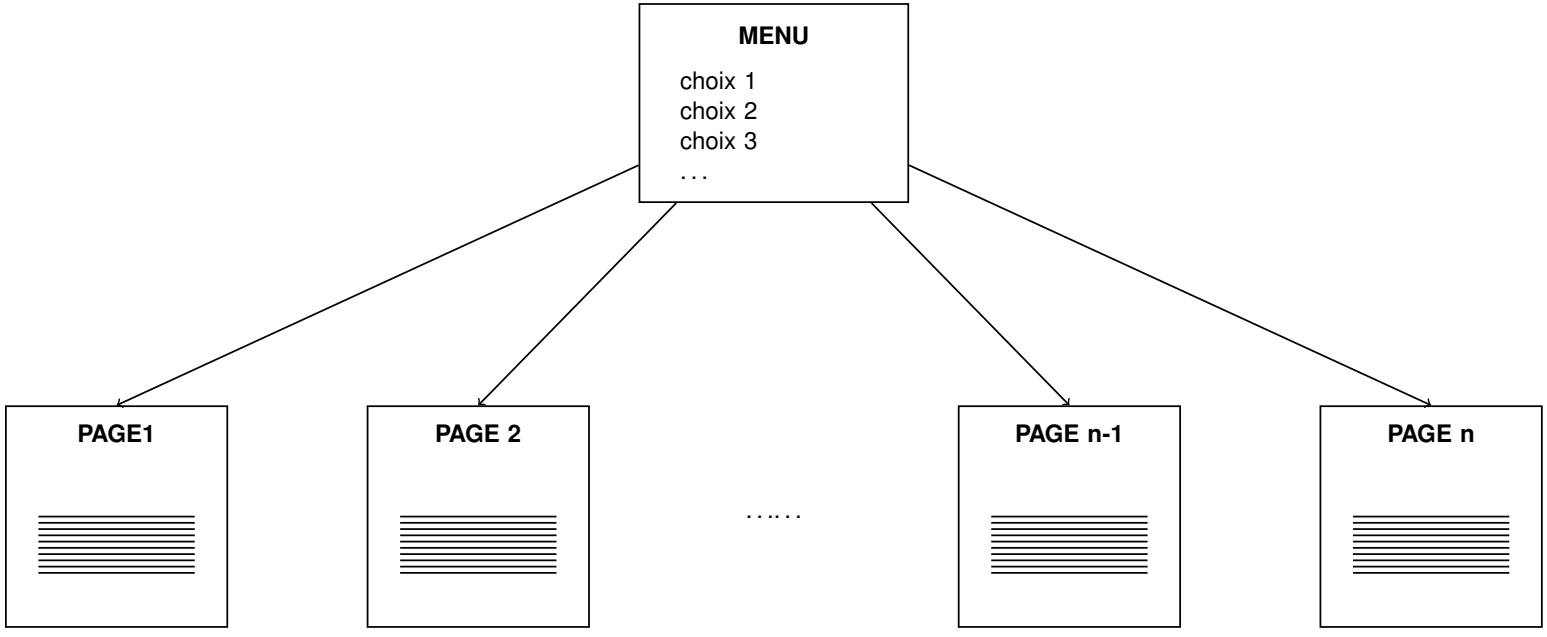
Authentification

Philippe Mathieu & Guillaume Dufrene

IUT-A Lille
http://www.iut-a.univ-lille.fr
prenom.nom@univ-lille.fr

Un site web “classique”

Un menu qui donne accès à une arborescence de pages



Evidemment, tel quel, toutes les pages sont accessibles par tout le monde !

Limiter les accès

Université de Lille

On souhaite que ces pages ne soient plus accessibles à tous mais uniquement aux personnes autorisées

1. La liste des personnes autorisées doit être établie
2. On ajoute dans une base de données une table des login/mdp autorisés : `users(login,mdp)`
3. Celui qui est présent dans cette table peut accéder au menu et autres pages, celui qui n'y est pas doit etre rejeté !

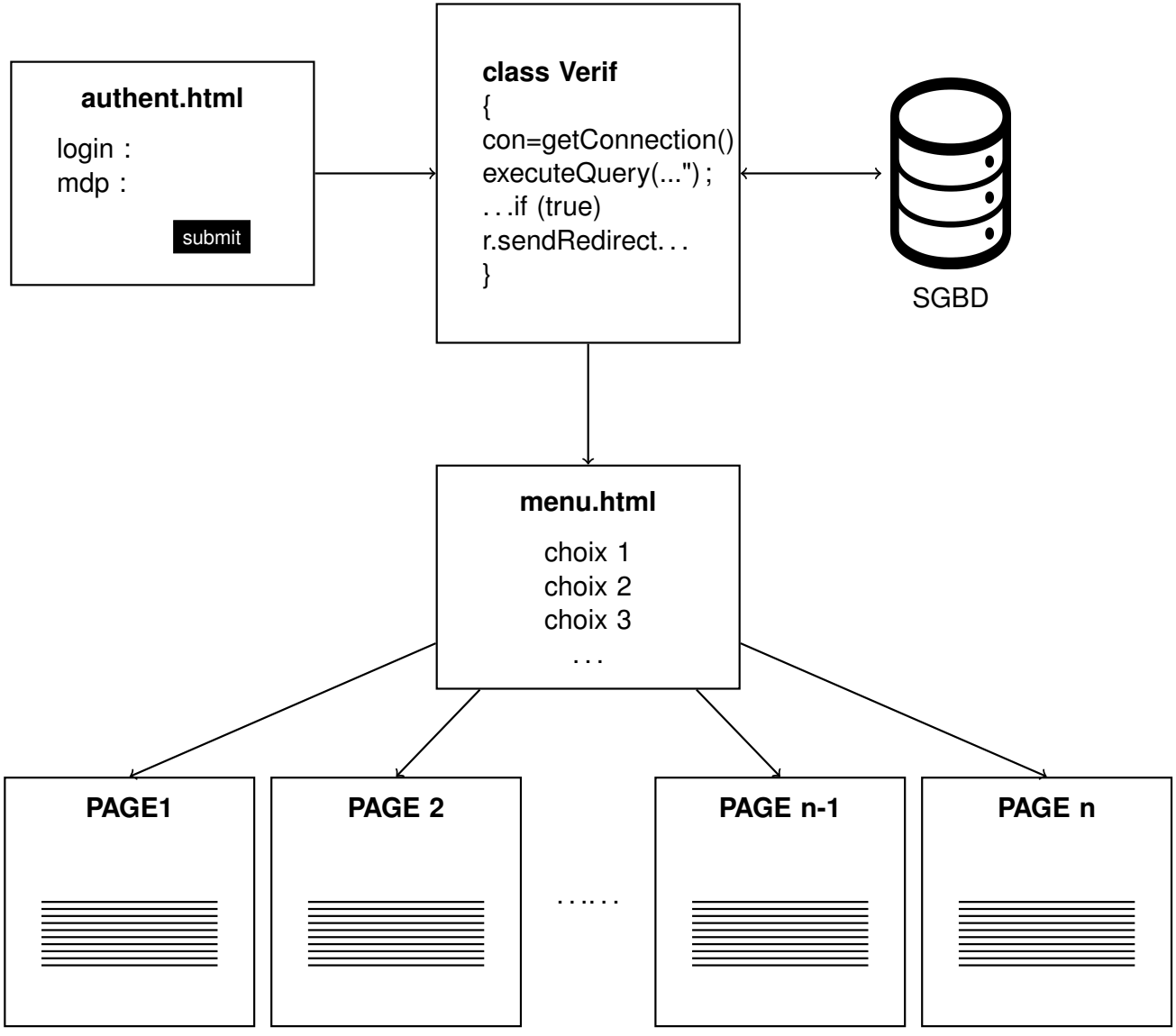
Principe de fonctionnement

Université de Lille

- Pour authentifier les personnes on crée une page `ident.html` qui contient un formulaire de saisie du login/mdp
- Ce formulaire appelle une servlet `Verif` qui
  1. Effectue une requête à la base de données
  2. Vérifie si le login/mdp existe dans `users`
  3. Redirige selon le cas vers le `menu.html` ou vers une page d'`erreur`

Principe de fonctionnement

Université de Lille



Forcer la connection de l'utilisateur

Université de Lille

Malheureusement on accède encore aux pages via leurs URL !

- Un utilisateur bienveillant peut "bookmarker" une page et y revenir ensuite, en n'étant alors plus identifié.
- Un utilisateur malveillant peut "shunter" l'authentification pour accéder aux ressources sans être identifié.

Il faut **forcer** l'utilisateur à passer par la page d'authentification
Toute tentative d'y échapper doit ramener à l'authentification.

Utilisation d'une session

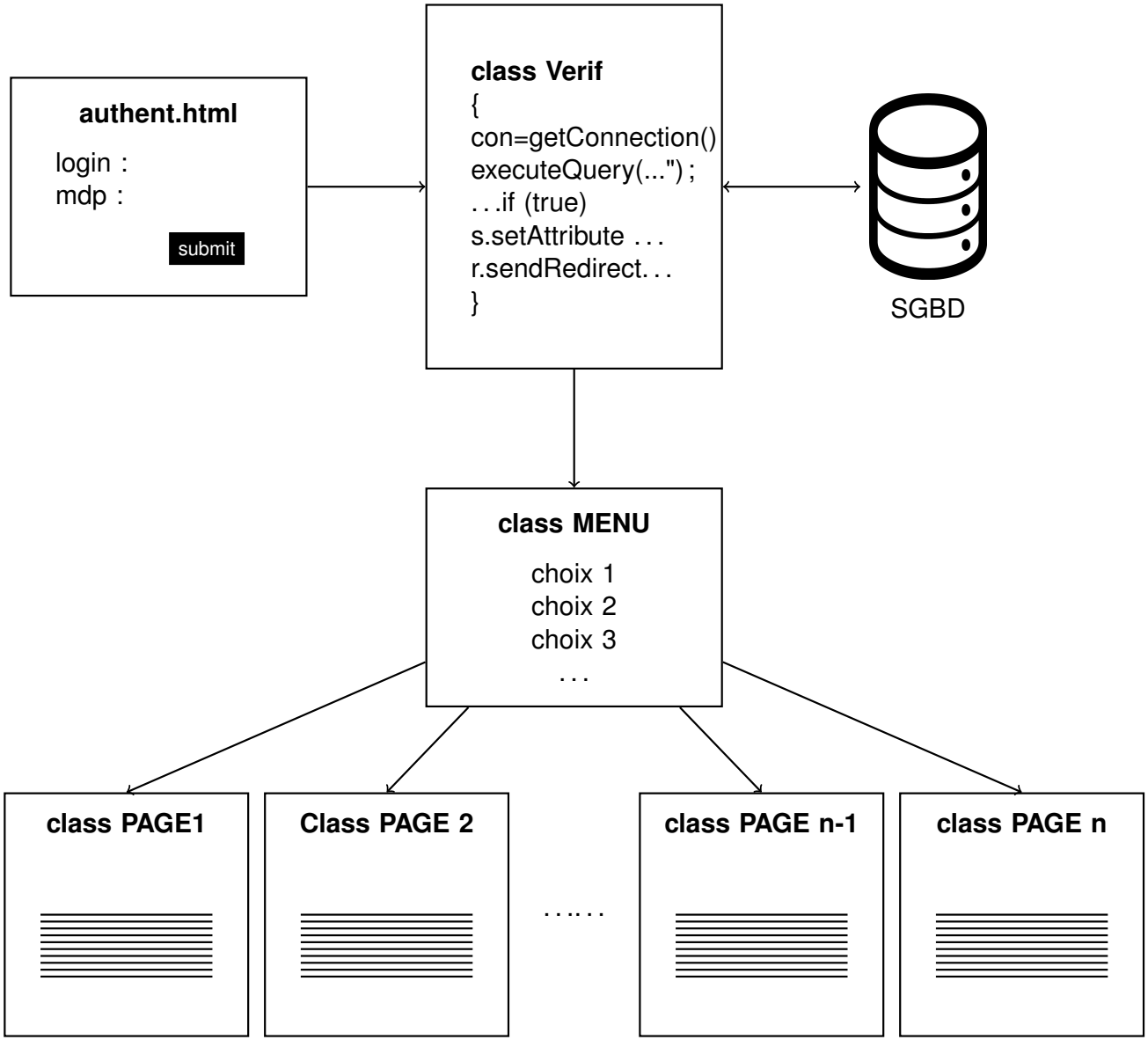
Université de Lille

- Dès qu'un utilisateur se connecte, la servlet `verif` range "une marque" dans la session utilisateur
- Toutes les autres pages sont transformées en servlets
- Pour toutes les autres pages, on vérifie si la marque est bien présente
  - elle n'est pas présente : l'utilisateur ne s'est pas loggué ; on le renvoie sur la page `ident.html` (méthode `sendRedirect`)
  - elle est présente : on continue la page normalement

N'importe quelle "marque" peut convenir.
En général c'est le login qui est utilisé

Les pages sont maintenant des classes

Université de Lille



Une première version

Université de Lille

Dans la servlet Verif

```
// accès à la base et test de la réponse
// ...
HttpSession session = request.getSession(true);
session.setAttribute("login",login);
response.sendRedirect("menu");
```

Dans toutes les autres pages ...

```
HttpSession session = request.getSession(true);
if (session.getAttribute("login")==null){
    response.sendRedirect("../authent.html");
    return; }
// reste de la page
```

Eviter de reparcourir tout le site

Université de Lille

- Si le site contient de nombreuses pages, il est important de laisser l'utilisateur "bookmarker" des pages
- La solution précédente oblige à reparcourir tout le site après authentification
- Celui qui rappelle une page, n'est pas systématiquement malveillant, il doit pouvoir s'authentifier et y accéder directement
- D'autant plus important que le site contient de nombreuses pages

Retour direct à la page appelée

Université de Lille

Dans toutes les pages PageX

```
HttpSession session = request.getSession(true);
if (session.getAttribute("login")==null) {
  response.sendRedirect("Authent?origine=PageX");
  return; }
// reste de la page
```

la page d'authentification devient une servlet

- Appel de `Verif` à partir d'un formulaire login/mdp.
- Ajout d'un champs caché `origine` dans le formulaire

Dans la servlet Verif

```
session.setAttribute("login",login);
response.sendRedirect(request.getParameter("origine"))
```

En résumé

Université de Lille

- Dès que des données personnelles sont utilisées, une authentification est nécessaire
- La technique classique consiste à utiliser un marqueur dans la session
- Toutes les pages vérifient la présence de ce marqueur et refusent l'accès s'il n'est pas présent.