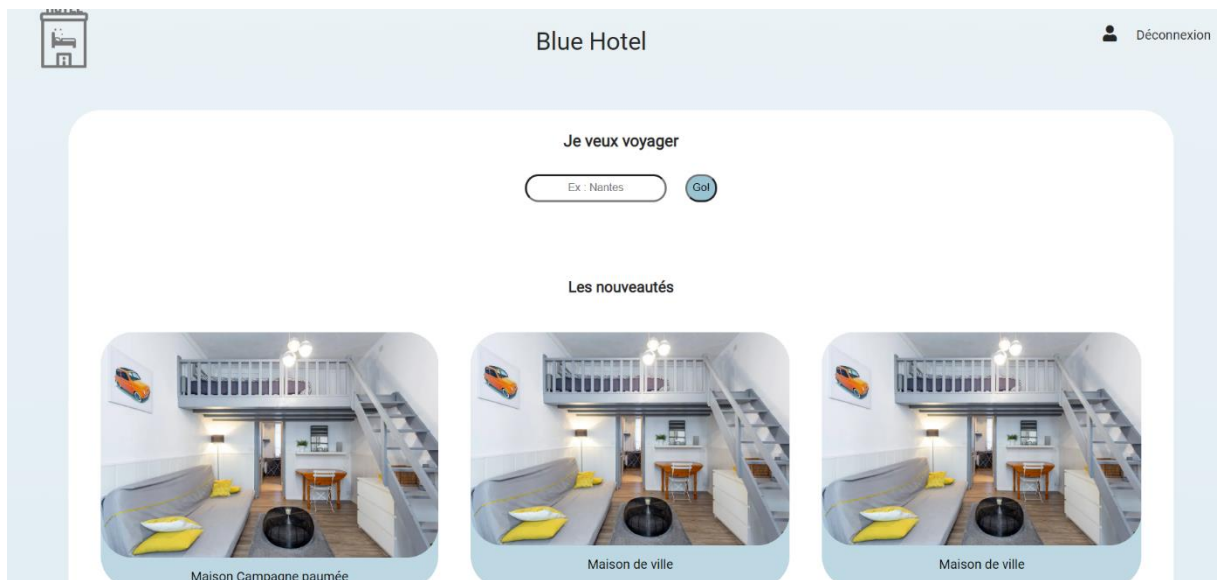


Suivi de projet 3

Le BLUE HOTEL

JACOB Aymeric

MECHIN Andréa



Sommaire

1. Architecture du site
2. Structure de la base de données
3. Fonctionnalités majeures
4. Technologies utilisées

1. Architecture du site :

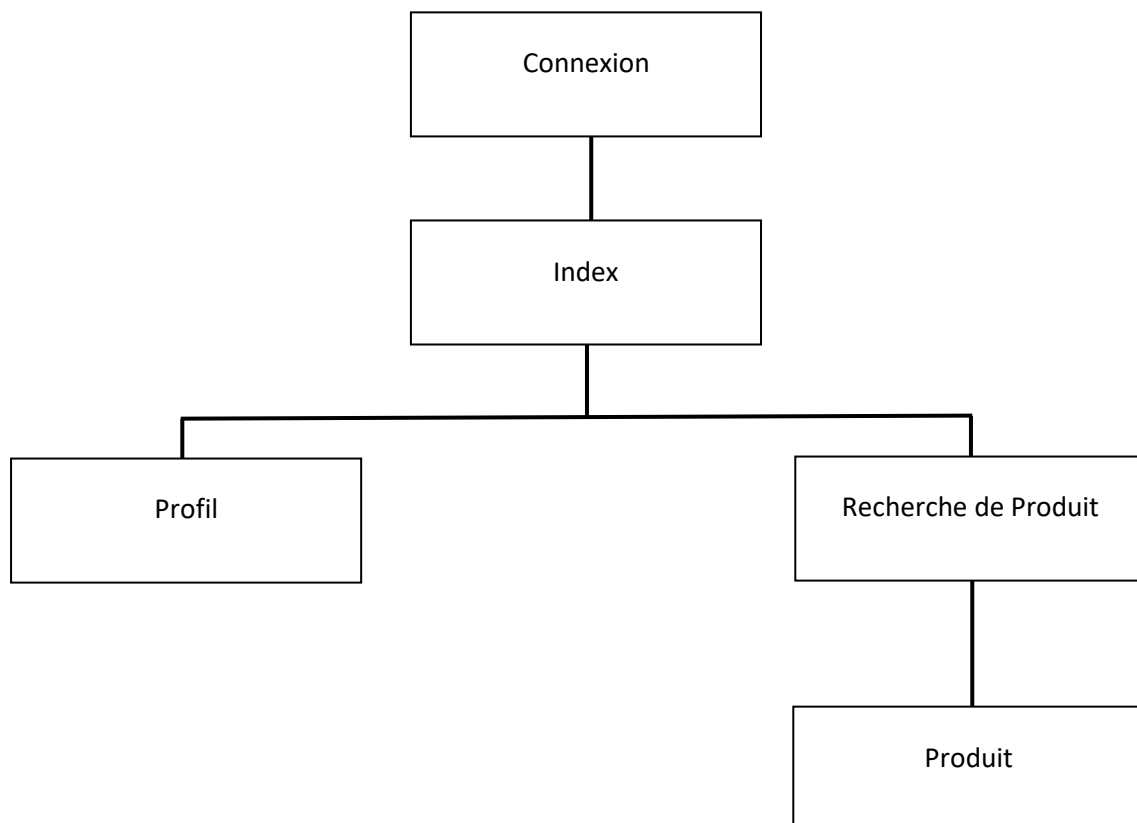
Pour l'architecture de notre site, nous avons fait le choix d'imposer la page de connexion / inscriptions avant de pouvoir avoir accès aux autres pages.

Une fois connecté, nous sommes redirigés vers l'index, à savoir la page d'accueil qui permet de faire une recherche en écrivant le lieu dans un formulaire.

La recherche étant lancée, nous arrivons sur la page correspondant aux locations de la ville en question.

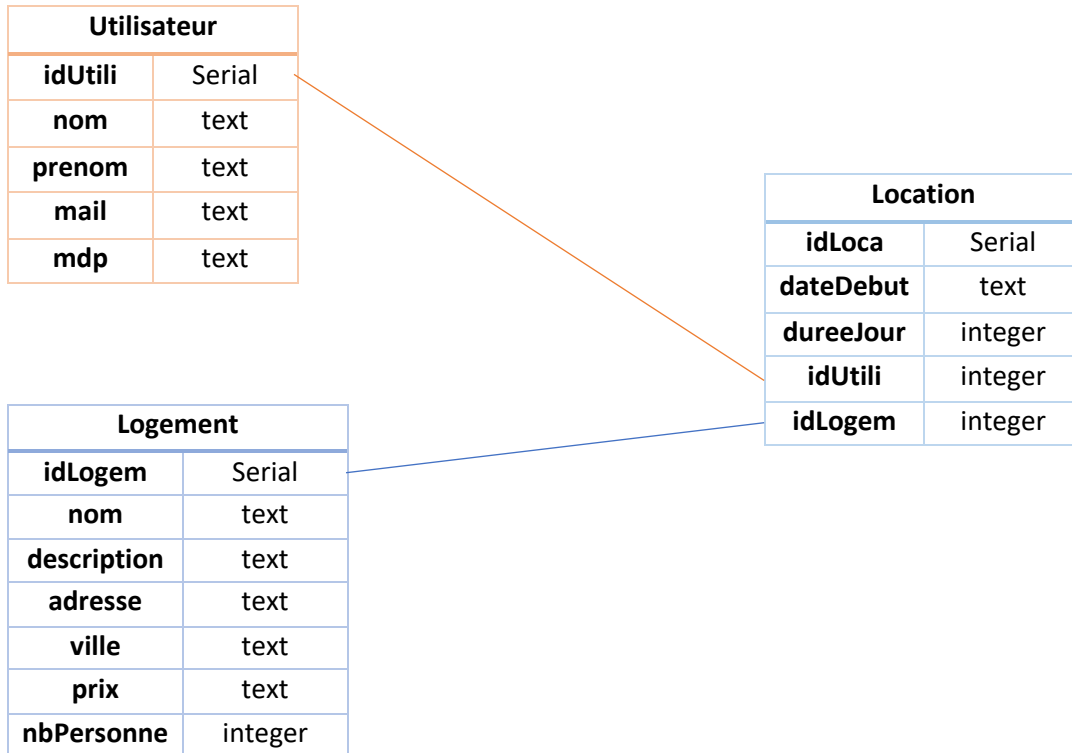
Cette page recherche fonctionne comme un catalogue, nous pouvons donc cliquer sur le bien qui nous intéresse, qui nous amène sur sa fiche de description avec son nom, sa description, son adresse, son prix à la nuit, ses dates disponibles et pour finir un bouton de réservation

D'autre part nous avons un icône profil, sur ce profil s'affiche nos informations ainsi que nos réservations, que nous pouvons supprimer.



2. Structure base de données :

Pour la base de données, nous avons voulu faire assez simple en ne créant pas énormément de tables, car nous avons estimé que pour ce projet les trois ci-dessous suffisaient amplement.



```
drop table if exists Utilisateur;
drop table if exists Logement;
drop table if exists Location;

Create table Utilisateur (
  idUtili Serial primary key,
  nom text,
  prenom text,
  mail text,
  mdp text
);

Create table Logement(
  idLogem Serial primary key,
  nom text,
  description text,
  adresse text,
  ville text,
  prix text,
  nbPersonne integer
);

Create table Location (
  idLoca Serial primary key,
  dateDebut text,
  dureeJour integer,
  idUtili integer,
  idLogem integer,
  foreign key (idUtili) references Utilisateur(idUtili),
  foreign key (idLogem) references Logement(idLogem)
);
```

3. Fonctionnalités majeures :

Fonction permettant de se connecter à la base de données :

```
function connectBDD() {
    $host = '127.0.0.1';
    $port = '3306';
    $db = 'blueHotel';
    $login = 'root';
    $password = '';

    try {
        $pdo = new PDO('mysql:host='.$host.';port='.$port.';dbname='.$db, $login, $password);
        $pdo->exec("set names utf8");
        $pdo->setAttribute(PDO::ATTR_ERRMODE, PDO::ERRMODE_EXCEPTION);
    } catch (PDOException $e) {
        var_dump($e->getMessage());
    }

    return $pdo;
}
```

Fonction qui gère la connexion via un compte :

```
function seConnecter($login, $mdp) {
    $pdo = connectBDD();

    $rows = $pdo->query("SELECT * FROM utilisateur WHERE mail='".$login."'");
    $error = "Pas de compte";
    $boolError = false;

    while ($row = $rows->fetch()){
        if($row['mail'] == $login && $row['mdp'] == $mdp) {
            //$_SESSION['logged'] = true;
            $_SESSION['id'] = $row['idUtili'];
            $_SESSION['login'] = $row['mail'];
            $error = "Bien connecté";
            break;
        }
        else {
            $error = "Mauvais mdp";
            $boolError = false;
        }
    }
    $rows->closeCursor();
    $pdo = null;
    return array($boolError, $error);
}
```

Et la réservation d'un logement :

```
$field = array('dateDebut', 'dureeJour', 'idUtili', 'idLogem');  
  
$value = array();  
array_push($value, $_POST['date']);  
array_push($value, $_POST['nbJour']);  
array_push($value, $_SESSION['id']);  
array_push($value, $_GET['id']);  
  
insertDB('location', $field, $value);  
  
$error = 'ma réservation est fé!!';
```

4. Technologies utilisées :

Pour ce projet nous avons utilisé le HTML/CSS pour le rendu visuel, pour ce qui est des animations côté client nous avons utilisé du JavaScript. Et afin de pouvoir gérer la base de données basée sur MySQL côté serveur, nous avons utilisé du PHP.

Quant à la gestion de projet, nous nous sommes servis de GIT et nous communiquons par Discord.