Projet : Conception, réalisation et développement d’une application web

Table des matières

[Présentation de l’entreprise ABRITEL: 3](#_Toc89813131)

[L’organigramme : 3](#_Toc89813132)

[Répartition des taches : 3](#_Toc89813133)

[Les cibles du business model: 3](#_Toc89813134)

[Les biens proposés par l’agence : 3](#_Toc89813135)

[Méthodologie de travail : 3](#_Toc89813136)

[Liste des scénarios de l’agence 4](#_Toc89813137)

[Soumission d’un bien immobilier 4](#_Toc89813138)

[Prospection d’un bien immobilier 5](#_Toc89813139)

[Validation d’un bien immobilier 5](#_Toc89813140)

[Mise en location d’un bien immobilier 5](#_Toc89813141)

[Modération des prix des biens immobilier 5](#_Toc89813142)

[Organisation des rendez-vous 5](#_Toc89813143)

[Contras de locations 6](#_Toc89813144)

[Modération des payements 6](#_Toc89813145)

[Suivi post location 6](#_Toc89813146)

[Automatisation du système d’information de l’agence : 6](#_Toc89813147)

[Flux d'information : 6](#_Toc89813148)

[Diagramme des flux : 7](#_Toc89813149)

[Dictionnaire de données : 8](#_Toc89813150)

[Model UML 10](#_Toc89813151)

[Modèle relationnel : 11](#_Toc89813152)

[Requêtes SQL 11](#_Toc89813153)

[Création de la base de données 11](#_Toc89813154)

[Peuplé une partie de la base de données avec des données test : 15](#_Toc89813155)

[Requêtes proposé : 18](#_Toc89813156)

[Technologies et outille de développement : 19](#_Toc89813157)

[Langage de programmation : 19](#_Toc89813158)

[Outille de développement 20](#_Toc89813159)

[SGBD : MYSQL 20](#_Toc89813160)

[Environnement développement : WAMP 20](#_Toc89813161)

[Framework de développement d’interface ccs : 20](#_Toc89813162)

[Moteur de versioning GIT : 20](#_Toc89813163)

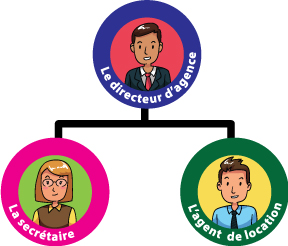
[Charte graphique : 21](#_Toc89813164)

[Imprime écran de l’application web 21](#_Toc89813165)

# Présentation de l’entreprise ABRITEL:

Abritel est une société qui œuvre dans l’accélération de transaction immobilière, principalement la location de maisons de vacances.

Afin de garantir des produits de qualité et un service client irréprochable Le système d’activité de l’entreprise passe par plusieurs étapes bien spécifiques, et chaque étape est divisé en plusieurs taches et cela pour garantir une synergie entre chaque acteur de la chaine de travail de la société.



## L’organigramme :

L’agence dispose d’un organigramme détaillé qui se divise en trois rôles :

* Le directeur d’agence
* La secrétaire
* L’agent de location

## Répartition des taches :

Chaque rôle dois faire des taches bien définit :

|  |  |  |
| --- | --- | --- |
| Le directeur d’agence | La secrétaire | L’agent de location |
| * Signe les contrats de location * Délivre ou encaisse les montants de location | * réception des clients | * Déplacements sur site * Prise de photos et vidéos * Présentation des produits aux demandeurs |

## Les cibles du business model:

Du fait que l’agence ARBITEL active dans l’accélération de transaction, leurs cibles principales sont :

* Les clients qui cherchent à louer un bien immobilier pour les vacances.
* Les clients propriétaire qui mettent en location leur bien immobilier.

## Les biens proposés par l’agence :

Les biens ciblé ou proposés par l’agence sont soit des appartements, des villas, des étages de villa, des maisons ou des bungalows.

## Méthodologie de travail :

Selon les cas, ABRITEL a mis en place différents scénarios ! Et chaque scénario prend des directives précises à suivre.

Les diffèrent cas font intervenir plusieurs rôles dans la chaine d’exécution de chaque scénario, et ce qui lie entre chaque élément ce sont les flux d’actions.

## Liste des scénarios de l’agence

### Soumission d’un bien immobilier

La soumission se fait par le **propriétaire du bien (qui lui peut être un client de l’agence)**, qui va envoyer les informations nécessaires demandé par l’agence à **la secrétaire** qui sont :

* Type du bien (appartements, villa, étage de villa, maison, bungalow)
* Description exact :
  + identificateur unique
  + adresse
  + wilaya
  + surface
  + nombre de pièce
  + une ou plusieurs photos de chaque pièce
* Les commodités externes
  + Distances entre le bien et les commerces
  + Les grandes surfaces
  + les stations de bus
  + les gares routières
  + les gares ferroviaires
  + les ports
  + l’aéroport
  + les salles de sport
  + les plages
  + hôpital
  + cabinet dentaire
* liste des équipements
  + Climatisation (qualité)
  + TV et récepteur satellite
  + Cuisinière,
  + Micro-onde,
  + Frigidaire,
  + Machine a lavé
  + Lave-vaisselle,
  + Tables/chaises
  + Ustensiles de cuisine
  + Literies
  + Chambre a couché (single, double...)

### Prospection d’un bien immobilier

La prospection du bien survient juste après une soumission de bien, **la secrétaire** passera le dossier a **l’agent de location** qui va ensuite se déplacé sur site pour vérifier les informations fournis par le soumissionnaire et ensuite fait un complément d’informations tel que :

* La prise de photos et de vidéos
* Situation géographique
* État des lieux
* État des équipements
* État du bien

### Validation d’un bien immobilier

La validation du bien se fera au niveau du **directeur de l’agence**, qui lui va juger selon les informations collectés par **l’agent de location** si « oui » ou « non » sont conforme au standard de l’agence.

### Mise en location d’un bien immobilier

Suite à la validation du bien par le **directeur**, le **soumissionnaire** signera un document qui comporte :

* l’exclusivité de location
* les modalités de payement
* le prix du bien
* la marge bénéficiaire brute
* les conditions du contrat

### Modération des prix des biens immobilier

La modération des prix des biens se font par le **directeur** et selon plusieurs critères :

* Commodités interne
* Commodités externe
* Etat du bien
* Localisation géographique
* Les saisons de mise en location

### Organisation des rendez-vous

L’organisation des rendez-vous entre le client et **l’agent de location** se fait chez la **secrétaire** qui va fixer les dates selon le planning des rendez-vous de **l’agent,** aussile client souhaitant visiter un bien devra signer un contrat qui protège l’agence en cas de fraude.

### Contras de locations

Les contras de locations se font chez le **directeur**, le contrat entre **l’agence** et le **client** se fera sous conditions :

* Payer une caution de protection contre la dégradation du bien
* Le respect de la période de location
* En cas de réservation lointaine les clients doit verser 30% du montant de location (en cas d’annulation, le montant avancé ne sera pas remboursé)

### Modération des payements

**Le directeur** gère les rentrées d’argent issue des **locations** et la **rémunération** des propriétaires de bien qui sont loué.

### Suivi post location

L’agent de location effectuera une visite post location pour déterminé l’état des lieux, ainsi il pourra déterminer si  « ou »i ou « non » la caution sera restauré à l’ancien locataire ou bien le dédommagement du propriétaire.

## Automatisation du système d’information de l’agence :

L’étude des actions de l’agence va nous permettre de tracé un diagramme des flux qui retrace toute les interactions qui sont nécessaire dans la numérisation du système d’information :

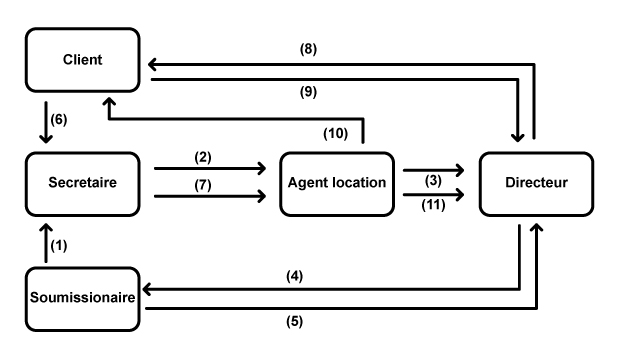
### Flux d'information :

Un flux suppose un transfert d'info entre deux ou plusieurs entités autour d'une séquence émetteur - récepteur. Les flux d'informations sont des échanges de contenu entre un ou plusieurs acteurs et un ou plusieurs objets. La nature des contenus est variable (contenu oral, procédure informatique, transition d'un objet - transport...). Ces échanges peuvent être matériels et immatériels. Les flux informationnels peuvent concerner des individus ou des procédures automatisées entre machines et robots. On parle d'échange d'informations, d'échange de données, d'échange d'instruction.

Dans notre cas nous allons traiter le transfert d’informations entre les cinq acteurs de l’entreprise ABRITEL qui rentre dans le cadre de leur activité d’accélération de transaction entre soumissionnaire et locataire de maisons de vacance, et cela va concerner le directeur d’agence, la secrétaire, l’agent de location, le soumissionnaire et le client final.

La procédure est de réalisé un schéma graphique qui comporte les liens numéroté d’actions qui ce passe entre chaque acteur, et indiqué par la suite le nom de chaque action et la direction de celle la suivant un ordre logique. Toute cette procédure détermine les actions interne et externe et le degré d’automatisation.

### Diagramme des flux :



|  |  |
| --- | --- |
| **Ordre** | **Action** |
| **1** | Dépôt d’annonce avec descriptions et informations nécessaire |
| **2** | Crée un RDV pour la visite du bien |
| **3** | Envoi le rapport pour l’approbation de l’annonce |
| **4** | Approbation et fixation des prix |
| **5** | Signature des clauses du contrat |
| **6** | Demande de RDV |
| **7** | Signature contrat de principe |
| **8** | Organise un RDV de visite |
| **9** | Contrat de location et montant du payement |
| **10** | Signature et payement |
| **11** | Etat des lieux poste location |
| **12** | Rapport état des lieux |
| **13** | Remboursement ou non de la caution suivant le rapport |

**Note à connaitre : Un client (sous loueur) peut être un soumissionnaire (un loueur) et un soumissionnaire peut aussi être un client, de ce fait la séparation dans le diagramme des flux entre les deux rôles sont fait pour mieux exprimé les relations entre les différentes entités, ce qui nous donne la possibilité de schématisé dans le diagramme MCD le client et le soumissionnaire en une seul et même entité avec les mêmes avantages et contraintes.**

### Dictionnaire de données :

|  |  |  |  |
| --- | --- | --- | --- |
| N° | **Code** | **Description** | **Type et longueur** |
| 1 | **Users** | **Table utilisateurs** |  |
| 2 | user\_id | **Identifient unique** | **PRIMARY KEY** |
| 3 | prenom | **Stockage des prénoms** | **VARCHAR 30** |
| 4 | nom | **Stockage des noms** | **VARCHAR 30** |
| 5 | adresse | **Stockage des adresses** | **VARCHAR 30** |
| 6 | email | **Stockage des emails** | **VARCHAR 30** |
| 7 | password | **Stockage des mots de passe** | **VARCHAR 30** |
| 8 | role | **Stockage des role (directeur, agent)** | **VARCHAR 30** |
| 9 | **Telephones** | **Table** |  |
| 10 | telephone\_id | **Identifient unique** | **PRIMARY KEY** |
| 11 | numero | **Stockage des numéros de téléphones** | **VARCHAR 30** |
| 12 | user\_id | **Clef étrangère référence table users** | **FOREIGN KEY** |
| 13 | **Contrats** | **Table** |  |
| 14 | contrat\_id | **Identifient unique** | **PRIMARY KEY** |
| 15 | contrat\_type | **Contient les types de contrat** | **VARCHAR 30** |
| 16 | date\_deb | **Date début du contrat** | **DATE** |
| 17 | date\_fin | **Date fin du contrat** | **DATE** |
| 18 | informations | **Informations du contrat** | **VARCHAR 30** |
| 19 | user\_id | **Clef étrangère référence table users** | **FOREIGN KEY** |
| 20 | an\_id | **Clef étrangère réf table annonces** | **FOREIGN KEY** |
| 21 | **Clauses** | **Table** |  |
| 22 | clause\_id | **Identifient unique** | **PRIMARY KEY** |
| 23 | clause\_nom | **Nom de la clause** | **VARCHAR 30** |
| 24 | clause\_montant | **Montant de la clause** | **VARCHAR 30** |
| 25 | contrat\_id | **Clef étrangère réf table contrats** | **FOREIGN KEY** |
| 26 | **Annonces** | **Table** |  |
| 27 | an\_id | **Identifient unique** | **PRIMARY KEY** |
| 28 | an\_statut | **Etat accepter ou non** | **VARCHAR 30** |
| 29 | an\_type | **Type d’annonce** | **VARCHAR 30** |
| 30 | an\_adresse | **Adresse de l’annonce** | **VARCHAR 30** |
| 31 | an\_surface | **Surface du bien** | **VARCHAR 30** |
| 32 | an\_piece | **Nombre de pièces du bien** | **NUMBER** |
| 33 | wilaya | **Wilaya du bien** | **VARCHAR 30** |
| 34 | user\_id | **Clef étrangère référence table users** | **FOREIGN KEY** |
| 35 | **RDVS** | **Table rendez-vous** |  |
| 36 | rdv\_id | **Identifient unique** | **PRIMARY KEY** |
| 37 | rdv\_date\_demander | **Date de rendez-vous demandé** | **DATE** |
| 38 | rdv\_date\_prevu | **Date prévu par l’agent** | **DATE** |
| 39 | rdv\_heure | **Heure du rendez-vous** | **TIME** |
| 40 | Statut | **Statut du rdv (en cour, annuler...)** | **VARCHAR 30** |
| 41 | user\_id | **Clef étrangère référence table users** | **FOREIGN KEY** |
| 42 | an\_id | **Clef étrangère réf table annonces** | **FOREIGN KEY** |
| 43 | **Reservations** | **Table** |  |
| 44 | reserv\_type | **Type de réservation** | **VARCHAR 30** |
| 45 | reserv\_id | **Identifient unique** | **PRIMARY KEY** |
| 46 | versement | **Montant versement (30% d’avance)** | **VARCHAR 30** |
| 47 | user\_id | **Clef étrangère référence table users** | **FOREIGN KEY** |
| 48 | an\_id | **Clef étrangère réf table annonces** | **FOREIGN KEY** |
| 49 | **Equipements** | **Table** |  |
| 50 | equip\_id | **Identifient unique** | **PRIMARY KEY** |
| 51 | equip\_type | **Type d’équipements** | **VARCHAR 30** |
| 52 | equip\_nom | **Nom de l’équipement** | **VARCHAR 30** |
| 53 | equip\_etat | **Etat de l’équipement** | **VARCHAR 30** |
| 54 | an\_id | **Clef étrangère réf table annonces** | **FOREIGN KEY** |
| 55 | **Medias** | **Table** |  |
| 56 | an\_id | **Clef étrangère réf table annonces** | **FOREIGN KEY** |
| 57 | media\_id | **Identifient unique** | **PRIMARY KEY** |
| 58 | media\_type | **Type vidéo ou photo** | **VARCHAR 30** |
| 59 | media\_dir | **Lien du dossier de destination** | **VARCHAR 50** |
| 60 | media\_descrip | **Description du media** | **VARCHAR 300** |
| 61 | media\_nom | **Nom du media** | **VARCHAR 30** |
| 62 | **Comodites** | **Table** |  |
| 63 | como\_id | **Identifient unique** | **PRIMARY KEY** |
| 64 | como\_nom | **Nom de la commodité** | **VARCHAR 30** |
| 65 | como\_type | **Type de la commodité (électro, meuble)** | **VARCHAR 30** |
| 66 | distance | **Distance entre le bien et la commodité** | **VARCHAR 30** |
| 67 | an\_id | **Clef étrangère réf table annonces** | **FOREIGN KEY** |
| 68 | **Ptarifaires** | **Table périodes tarifaire** |  |
| 69 | id\_periode | **Identifient unique** | **PRIMARY KEY** |
| 70 | couleurs | **Nom des périodes** | **VARCHAR 30** |
| 71 | date\_debut | **Date début de la période** | **DATE** |
| 72 | date\_fin | **Date fin de la période** | **DATE** |
| 73 | **Prix** | **Table** |  |
| 74 | prix | **Contient le prix de l’annonce** | **VARCHAR 30** |
| 75 | an\_id | **Clef étrangère réf table annonces** | **FOREIGN KEY** |
| 76 | id\_periode | **Clef étrangère réf table annonces** | **FOREIGN KEY** |

### Model UML

Diagramme réalisé avec l’outille STAR ULM, une copie du fichier original sera lié avec le document

### Modèle relationnel :

|  |  |
| --- | --- |
| **Les tables et leurs colonnes** | |
| **tables** | **colonnes** |
| **Users** | **(user\_id, prenom, nom, adresse, email, role,password)** |
| **Telephones** | **(telephone\_id, numero, user\_id)** |
| **Contrats** | **(contrat\_id, contrat\_type, date\_deb, date\_fin, informations, user\_id, an\_id)** |
| **Clauses** | **(clause\_id, clause\_nom, clause\_montant, contrat\_id)** |
| **Annonces** | **(an\_id, an\_type,** [**an\_statut**](http://localhost/phpmyadmin/sql.php)**, an\_adresse, an\_surface, an\_pièce, wilaya, user\_id)** |
| **RDVS** | **(rdv\_id, rdv\_date\_demander, rdv\_date\_prevu, rdv\_heure, statut, user\_id, an\_id)** |
| **Reservations** | **(reserv\_type, reserv\_id, versement, user\_id, an\_id)** |
| **Equipements** | **(equip\_id, equip\_type, equip\_nom, equip\_etat, an\_id)** |
| **Medias** | **(media\_id, media\_type, media\_dir, media\_descrip, media\_nom, an\_id)** |
| **Comodites** | **(como\_id, como\_nom, como\_type, distance, an\_id)** |
| **Ptarifaires** | **(id\_periode, couleurs, date\_debut, date\_fin)** |
| **Prix** | **(prix, an\_id, id\_periode)** |

# Requêtes SQL

## Création de la base de données

|  |  |  |
| --- | --- | --- |
| Étapes | Procédures | Commandes |
| 1 | Création base de données | CREATE DATABASE location; |
| 2 | Utilisation BDD location | USE location ; |
| 3 | Création de la table | DROP TABLE IF EXISTS users;  CREATE TABLE users  (user\_id int NOT NULL PRIMARY KEY AUTO\_INCREMENT,  nom varchar(30) NOT NULL,  prenom varchar(30) NOT NULL,  email varchar(30) NOT NULL,  adresse varchar(30) NOT NULL,  password varchar(40) NOT NULL,  role varchar(30) NOT NULL); |
| 4 | Création de la table | DROP TABLE IF EXISTS Telephones;  CREATE TABLE Telephones  (telephone\_id int NOT NULL PRIMARY KEY AUTO\_INCREMENT,  numero varchar(30) NOT NULL); |
| 5 | Création de la table | DROP TABLE IF EXISTS Contrats;  CREATE TABLE Contrats  (contrat\_id int NOT NULL PRIMARY KEY AUTO\_INCREMENT,  Contrat\_type varchar(30) NOT NULL,  Date\_deb date NOT NULL,  Date\_fin date NOT NULL,  informations varchar(30) NOT NULL); |
| 6 | Création de la table | DROP TABLE IF EXISTS Clauses;  CREATE TABLE Clauses  (clause\_id int NOT NULL PRIMARY KEY AUTO\_INCREMENT,  caluse\_nom varchar(30) NOT NULL,  clause\_montant varchar(30) NOT NULL); |
| 7 | Création de la table | DROP TABLE IF EXISTS Annonces;  CREATE TABLE annonces  (an\_id int NOT NULL PRIMARY KEY AUTO\_INCREMENT,  an\_type varchar(30) NOT NULL,  an\_adresse varchar(30) NOT NULL,  an\_surface varchar(30) NOT NULL,  an\_pièce int NOT NULL,  wilaya varchar(30) NOT NULL); |
| 8 | Création de la table | DROP TABLE IF EXISTS Reservations;  CREATE TABLE Reservations  (reserv\_id int NOT NULL PRIMARY KEY AUTO\_INCREMENT,  reserv\_type varchar(30) NOT NULL,  versement varchar(30) NOT NULL); |
| 9 | Création de la table | DROP TABLE IF EXISTS rdvs;  CREATE TABLE rdvs  (rdv\_id int NOT NULL PRIMARY KEY AUTO\_INCREMENT,  rdv\_date\_prevu date NOT NULL,  rdv\_date\_demander date NOT NULL,  rdv\_heure time NOT NULL,  statut varchar(30) NOT NULL); |
| 10 | Création de la table | DROP TABLE IF EXISTS Ptarifaires;  CREATE TABLE Ptarifaires  (id\_periode int NOT NULL PRIMARY KEY AUTO\_INCREMENT,  couleurs varchar(30) NOT NULL,  date\_debut date NOT NULL,  date\_fin date NOT NULL); |
| 11 | Création de la table | DROP TABLE IF EXISTS Prix;  CREATE TABLE Prix  (prix numeric(19,4) NOT NULL); |
| 12 | Création de la table | DROP TABLE IF EXISTS Commodites;  CREATE TABLE Commodites  (como\_id int NOT NULL PRIMARY KEY AUTO\_INCREMENT,  Como\_type varchar(10) NOT NULL,  Como\_nom varchar(30) NOT NULL,  distance varchar(30) NOT NULL); |
| 13 | Création de la table | DROP TABLE IF EXISTS Equipements;  CREATE TABLE Equipements  (equip\_id int NOT NULL PRIMARY KEY AUTO\_INCREMENT,  equip\_type varchar(30) NOT NULL,  equip\_nom varchar(30) NOT NULL); |
| 14 | Création de la table | DROP TABLE IF EXISTS Medias;  CREATE TABLE Medias  (media\_id int NOT NULL PRIMARY KEY AUTO\_INCREMENT,  Media\_type varchar(30) NOT NULL,  Media\_nom varchar(30) NOT NULL,  Media\_dir varchar(50) NOT NULL,  Media\_descrip varchar(300) NOT NULL); |
| 15 | Modifier la table | ALTER TABLE Telephones  ADD COLUMN user\_id int; |
| 16 | Modifier la table | ALTER TABLE Telephones  ADD CONSTRAINT fk1  FOREIGN KEY(user\_id)  REFERENCES users(user\_id); |
| 17 | Modifier la table | ALTER TABLE contrats  ADD COLUMN user\_id int; |
| 18 | Modifier la table | ALTER TABLE Contrats  ADD CONSTRAINT fk2  FOREIGN KEY(user\_id)  REFERENCES userss(user\_id); |
| 19 | Modifier la table | ALTER TABLE contrats  ADD COLUMN an\_id int; |
| 20 | Modifier la table | ALTER TABLE Contrats  ADD CONSTRAINT fk3  FOREIGN KEY(an\_id)  REFERENCES Annonces(an\_id); |
| 21 | Modifier la table | ALTER TABLE clauses  ADD COLUMN contrat\_id int; |
| 22 | Modifier la table | ALTER TABLE Clauses  ADD CONSTRAINT fk4  FOREIGN KEY(contrat\_id)  REFERENCES Contrats(contrat\_id); |
| 23 | Modifier la table | ALTER TABLE annonces  ADD COLUMN user\_id int; |
| 24 | Modifier la table | ALTER TABLE Annonces  ADD CONSTRAINT fk5  FOREIGN KEY(user\_id)  REFERENCES userss(user\_id); |
| 25 | Modifier la table | ALTER TABLE reservations  ADD COLUMN user\_id int; |
| 26 | Modifier la table | ALTER TABLE Reservations  ADD CONSTRAINT fk6  FOREIGN KEY(user\_id)  REFERENCES userss(id\_users); |
| 27 | Modifier la table | ALTER TABLE reservations  ADD COLUMN an\_id int; |
| 28 | Modifier la table | ALTER TABLE Reservations  ADD CONSTRAINT fk7  FOREIGN KEY(an\_id)  REFERENCES Annonces(an\_id); |
| 29 | Modifier la table | ALTER TABLE rdvs  ADD COLUMN user\_id int; |
| 30 | Modifier la table | ALTER TABLE rdvs  ADD CONSTRAINT fk8  FOREIGN KEY(user\_id)  REFERENCES userss(user\_id); |
| 31 | Modifier la table | ALTER TABLE rdvs  ADD COLUMN an\_id int; |
| 32 | Modifier la table | ALTER TABLE rdvs  ADD CONSTRAINT fk9  FOREIGN KEY(an\_id)  REFERENCES Annonces(an\_id); |
| 33 | Modifier la table | ALTER TABLE prix  ADD COLUMN id\_periode int; |
| 34 | Modifier la table | ALTER TABLE Prix  ADD CONSTRAINT fk10  FOREIGN KEY(id\_periode)  REFERENCES Ptarifaires(id\_periode); |
| 35 | Modifier la table | ALTER TABLE prix  ADD COLUMN an\_id int; |
| 36 | Modifier la table | ALTER TABLE Prix  ADD CONSTRAINT fk11  FOREIGN KEY(an\_id)  REFERENCES Annonces(an\_id); |
| 37 | Modifier la table | ALTER TABLE commodites  ADD COLUMN an\_id int; |
| 38 | Modifier la table | ALTER TABLE Commodites  ADD CONSTRAINT fk12  FOREIGN KEY(an\_id)  REFERENCES Annonces(an\_id); |
| 39 | Modifier la table | ALTER TABLE equipements  ADD COLUMN an\_id int; |
| 40 | Modifier la table | ALTER TABLE Equipements  ADD CONSTRAINT fk13  FOREIGN KEY(an\_id)  REFERENCES Annonces(an\_id); |
| 41 | Modifier la table | ALTER TABLE medias  ADD COLUMN an\_id int; |
| 42 | Modifier la table | ALTER TABLE Medias  ADD CONSTRAINT fk14  FOREIGN KEY(an\_id)  REFERENCES Annonces(an\_id); |

## Peuplé une partie de la base de données avec des données test :

|  |  |
| --- | --- |
| tables | Requêtes |
| users | INSERT INTO `users` (`user\_id`, `nom`, `prenom`, `email`, `adresse`, `password`, `role`)  VALUES  ('1', 'zakaria', 'boucetta', 'zakaria-boucetta@mail.com', 'adresse', '123456789azerty', 'directeur'),  ('2', 'oussama', 'bendada', 'oussama-bendada@mail.com', 'adresse', '123456789azerty', 'agent'),  ('3', 'manel', 'talamali', 'manel-talamali@mail.com', 'adresse', '123456789azerty', 'client'),  ('4', 'amina', 'derf', 'amina-derf@mail.com', 'adresse', '123456789azerty', 'client'),  ('5', 'rym', 'tigiwart', 'rym-tigiwart@mail.com', 'adresse', '123456789azerty', 'client'),  ('6', 'feriel', 'bensalem', 'feriel-bensalem@mail.com', 'adresse', '123456789azerty', 'client'),  ('7', 'mohamed', 'bouarouj', 'mohamed-bouarouj@mail.com', 'adresse', '123456789azerty', 'client'),  ('8', 'toufik', 'slimani', 'toufik-slimani@mail.com', 'adresse', '123456789azerty', 'client'),  ('9', 'houria', 'hakem', 'houria-hakem@mail.com', 'adresse', '123456789azerty', 'client'); |
| telephones | INSERT INTO `telephones`(`telephone\_id`, `numero`, `user\_id`)  VALUES  (1,'+213 123 456 789',1),  (2,'+213 123 456 789',2),  (3,'+213 123 456 789',3),  (4,'+213 123 456 789',4),  (5,'+213 123 456 789',5),  (6,'+213 123 456 789',6),  (7,'+213 123 456 789',7),  (8,'+213 123 456 789',8),  (9,'+213 123 456 789',9); |
| annonces | INSERT INTO `annonces`(`an\_id`, `an\_type`, `an\_adresse`, `an\_surface`, `an\_pièce`, `wilaya`, `user\_id`, `an\_statut`)  VALUES  (1,'villa','adresse 12345','300',5,'alger',3,'approved'),  (2,'villa','adresse 12345','300',5,'alger',4,'approved'),  (3,'villa','adresse 12345','300',5,'alger',5,'approved'),  (4,'villa','adresse 12345','300',5,'alger',6,'approved'),  (5,'villa','adresse 12345','300',5,'alger',7,'approved'),  (6,'villa','adresse 12345','300',5,'alger',8,'approved'),  (7,'villa','adresse 12345','300',5,'alger',9,'approved'); |
| ptarifaires | INSERT INTO `ptarifaires`(`id\_periode`, `couleurs`, `date\_debut`, `date\_fin`)  VALUES  (1,'haute saison','2021-12-08','2022-03-08'),  (2,'basse saison','2022-03-08','2022-05-08'),  (3,'moyenne saison','2022-05-08','2022-06-08'),  (4,'normal saison','2022-06-08','2022-07-08'),  (5,'forte saison','2022-07-08','2022-08-08'); |
| prix | INSERT INTO `prix`(`prix`, `id\_periode`, `an\_id`)  VALUES  (5000,1,1),  (5000,1,2),  (5000,1,3),  (5000,1,4),  (5000,1,5),  (5000,1,6),  (5000,1,7),  (5000,1,8),  (5000,1,9), |
| medias | INSERT INTO `medias`(`media\_id`, `Media\_type`, `Media\_nom`, `Media\_dir`, `Media\_descrip`, `an\_id`)  VALUES  (1,'photo','photo.jpg','/dev/images/','description',1),  (2,'photo','photo.jpg','/dev/images/','description',2),  (3,'photo','photo.jpg','/dev/images/','description',3),  (4,'photo','photo.jpg','/dev/images/','description',4),  (5,'photo','photo.jpg','/dev/images/','description',5),  (6,'photo','photo.jpg','/dev/images/','description',6),  (7,'photo','photo.jpg','/dev/images/','description',7),  (8,'photo','photo.jpg','/dev/images/','description',8),  (9,'photo','photo.jpg','/dev/images/','description',9); |
| equipements | INSERT INTO `equipements`(`equip\_id`, `equip\_type`, `equip\_nom`, `an\_id`)  VALUES  (1,'electromenager','robot',1),  (2,'electromenager','robot',2),  (3,'electromenager','robot',3),  (4,'electromenager','robot',4),  (5,'electromenager','robot',5),  (6,'electromenager','robot',6),  (7,'electromenager','robot',7),  (8,'electromenager','robot',8),  (9,'electromenager','robot',9); |
| commodites | INSERT INTO `commodites`(`como\_id`, `Como\_type`, `Como\_nom`, `distance`, `an\_id`)  VALUES  (1,'hopital','mustapha','500',1),  (2,'hopital','mustapha','600',2),  (3,'hopital','mustapha','700',3),  (4,'hopital','mustapha','800',4),  (5,'hopital','mustapha','900',5),  (6,'hopital','mustapha','200',6),  (7,'hopital','mustapha','100',7),  (8,'hopital','mustapha','300',8),  (9,'hopital','mustapha','400',9); |
| contrats | INSERT INTO `contrats`(`contrat\_id`, `Contrat\_type`, `Date\_deb`, `Date\_fin`, `informations`, `user\_id`, `an\_id`)  VALUES  (1,'exclusivité','2020-12-12','2025-12-12','contract classé',3,1),  (2,'exclusivité','2020-12-12','2025-12-12','contract classé',4,2),  (3,'exclusivité','2020-12-12','2025-12-12','contract classé',5,3),  (4,'exclusivité','2020-12-12','2025-12-12','contract classé',6,4),  (5,'exclusivité','2020-12-12','2025-12-12','contract classé',7,5),  (6,'exclusivité','2020-12-12','2025-12-12','contract classé',8,6),  (7,'exclusivité','2020-12-12','2025-12-12','contract classé',9,7); |

\*une copie de la base de données peupler sera jointe à ce document.

## Requêtes proposé :

|  |  |  |
| --- | --- | --- |
| **Numero** | **Phrases** | Requêtes |
| 1 | Afficher les toutes les annonces approuvé | SELECT \* FROM  ((((annonces a  inner join medias m on a.an\_id = m.an\_id)  inner join equipements e on a.an\_id = e.an\_id)  inner join commodites c on a.an\_id = c.an\_id)  inner join prix p on a.an\_id = p.an\_id)  where an\_statut = 'approved' |
| 2 | Afficher les informations d’un utilisateur avec son numero de téléphone | SELECT \* FROM `users`  inner join telephones t  on users.user\_id = t.user\_id  WHERE users.user\_id = 1 |
| 3 | Afficher la liste des produits accepté qui appartienne à un client | SELECT \* FROM  (((((annonces a  inner join medias m on a.an\_id = m.an\_id)  inner join equipements e on a.an\_id = e.an\_id)  inner join commodites c on a.an\_id = c.an\_id)  inner join prix p on a.an\_id = p.an\_id)  inner join users u on u.user\_id = a.user\_id)  where  an\_statut = 'approved'  and  u.user\_id = 4 |
| 4 | afficher les annonces qui ont un prix minimum et une wilaya et une date libre spécifique | [SELECT](http://localhost/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/5.5/en/select.html) \* FROM ((((((annonces a inner join medias m on a.an\_id = m.an\_id) inner join equipements e on a.an\_id = e.an\_id) inner join commodites c on a.an\_id = c.an\_id) inner join prix p on a.an\_id = p.an\_id) inner join users u on u.user\_id = a.user\_id) inner join contrats co on co.an\_id = a.an\_id) where an\_statut = 'approved' [and](http://localhost/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/5.5/en/logical-operators.html#operator_and) p.prix < 300 [and](http://localhost/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/5.5/en/logical-operators.html#operator_and) a.wilaya = 'alger' [and](http://localhost/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/5.5/en/logical-operators.html#operator_and) co.date\_deb <> '2020-12-10' |
| 5 | afficher les annonce qui ont un prix minimum et maximum | [SELECT](http://localhost/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/5.5/en/select.html) \* FROM ((((((annonces a inner join medias m on a.an\_id = m.an\_id) inner join equipements e on a.an\_id = e.an\_id) inner join commodites c on a.an\_id = c.an\_id) inner join prix p on a.an\_id = p.an\_id) inner join users u on u.user\_id = a.user\_id) inner join contrats co on co.an\_id = a.an\_id) where an\_statut = 'approved' [and](http://localhost/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/5.5/en/logical-operators.html#operator_and) p.prix < 300 and p.prix > 500 |
| 6 | afficher le montant global des locations (la somme des rentrées d’argent) | [SELECT](http://localhost/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/5.5/en/select.html) [SUM](http://localhost/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/5.5/en/group-by-functions.html#function_sum)(prix) as sum\_prix FROM prix |
| 7 | afficher tous les RDV non traités par l'agent | SELECT \* FROM  RDVS  where statut = 'non traité' |
| 8 | Afficher l’utilisateur avec le role directeur | SELECT \* FROM  users  where role = ‘directeur’ |
| 9 | Afficher le numero de téléphone du directeur | SELECT t.numero FROM  users u inner join telephones t  on u.user\_id = t.user\_id  where role = 'directeur' |
| 10 | Afficher les périodes tarifaires | SELECT couleurs FROM ptarifaires |

# Technologies et outille de développement :

## Langage de programmation :

Le langage de programmation choisi et le **PHP 7.4**, le choix de ce langage et sa vitesse d’exécution et sa compatibilité parfaite avec les autres technologies de développements choisi pour ce projet, sa popularité en fait un atout avec une documentation riche et détaillé et sa retro compatibilité, PHP est un langage open source et gratuit.

**JavaScript :** ce langage est utilisé dans notre projet pour animé les segments de page et ainsi avoir du contenu dynamique, le JavaScript et un langage populaire et il existe dans la plupart des sites web dans le monde, une large documentation est disponible avec plein d’information de débogage, JavaScript est un langage open source et gratuit.

**HTML/CSS :** la base de la structure du site internet est faite avec ces deux scripts, ils représentent le squelette et la peau de notre site internet, donc il est essentiel de les utiliser dans notre projet.

## Outille de développement

### SGBD : MYSQL

Est un [système de gestion de bases de données](https://fr.wikipedia.org/wiki/Syst%C3%A8me_de_gestion_de_base_de_donn%C3%A9es) relationnelles (SGBDR). Il est distribué sous une double licence [GPL](https://fr.wikipedia.org/wiki/Licence_publique_g%C3%A9n%C3%A9rale_GNU) et [propriétaire](https://fr.wikipedia.org/wiki/Logiciel_propri%C3%A9taire). Il fait partie des logiciels de gestion de [base de données](https://fr.wikipedia.org/wiki/Base_de_donn%C3%A9es) les plus utilisés au monde, autant par le grand public (applications web principalement) que par des professionnels, en concurrence avec [Oracle](https://fr.wikipedia.org/wiki/Oracle_Database), [PostgreSQL](https://fr.wikipedia.org/wiki/PostgreSQL) et [Microsoft SQL Server](https://fr.wikipedia.org/wiki/Microsoft_SQL_Server).

L’utilisation de MySQL dans ce projet est directement lier a sa compatibilité avec PHP car la facilité de connecté le langage PHP et MySQL est facile et rapide et ne consomme pas de ressource, aussi il offre la robustesse accrue, surtout la plupart des serveurs web dans les hébergeurs utilise MySQL.

### Environnement développement : WAMP

WAMP est l’**abrégé de Windows, Apache, MySQL, PHP**. Il est dérivé de LAMP, qui signifie Linux, Apache, MySQL et PHP. Comme son nom l’indique, alors que **LAMP est utilisé sur des serveurs Linux, WAMP est utilisé sur des serveurs Windows**.

### Framework de développement d’interface ccs :

Source : <https://mdbootstrap.com/>

Bootstrap est une [collection d'outils](https://fr.wikipedia.org/wiki/Framework) utiles à la création du design (graphisme, animation et interactions avec la page dans le navigateur, etc.) de [sites](https://fr.wikipedia.org/wiki/Site_web) et d'[applications web](https://fr.wikipedia.org/wiki/Application_web). C'est un ensemble qui contient des codes [HTML](https://fr.wikipedia.org/wiki/HTML) et [CSS](https://fr.wikipedia.org/wiki/Cascading_Style_Sheet), des formulaires, boutons, outils de navigation et autres éléments interactifs, ainsi que des extensions [JavaScript](https://fr.wikipedia.org/wiki/JavaScript) en option. C'est l'un des projets les plus populaires sur la plate-forme de gestion de développement [GitHub](https://fr.wikipedia.org/wiki/GitHub).

### Moteur de versioning GIT :

Source : https://www.atlassian.com/fr/git/tutorials/what-is-git

Git est de loin le système de contrôle de version le plus largement utilisé aujourd'hui. Git est un projet open source avancé, qui est activement maintenu. À l'origine, il a été développé en 2005 par Linus Torvalds, le créateur bien connu du noyau du système d'exploitation Linux. De plus en plus de projets logiciels reposent sur Git pour le contrôle de version, y compris des projets commerciaux et en open source. Les développeurs qui travaillent avec Git sont bien représentés dans le pool de talents disponible, et la solution fonctionne bien sur une vaste gamme de systèmes d'exploitation et d'environnements de développement intégrés (IDE).

# Charte graphique :

## Imprime écran de l’application web

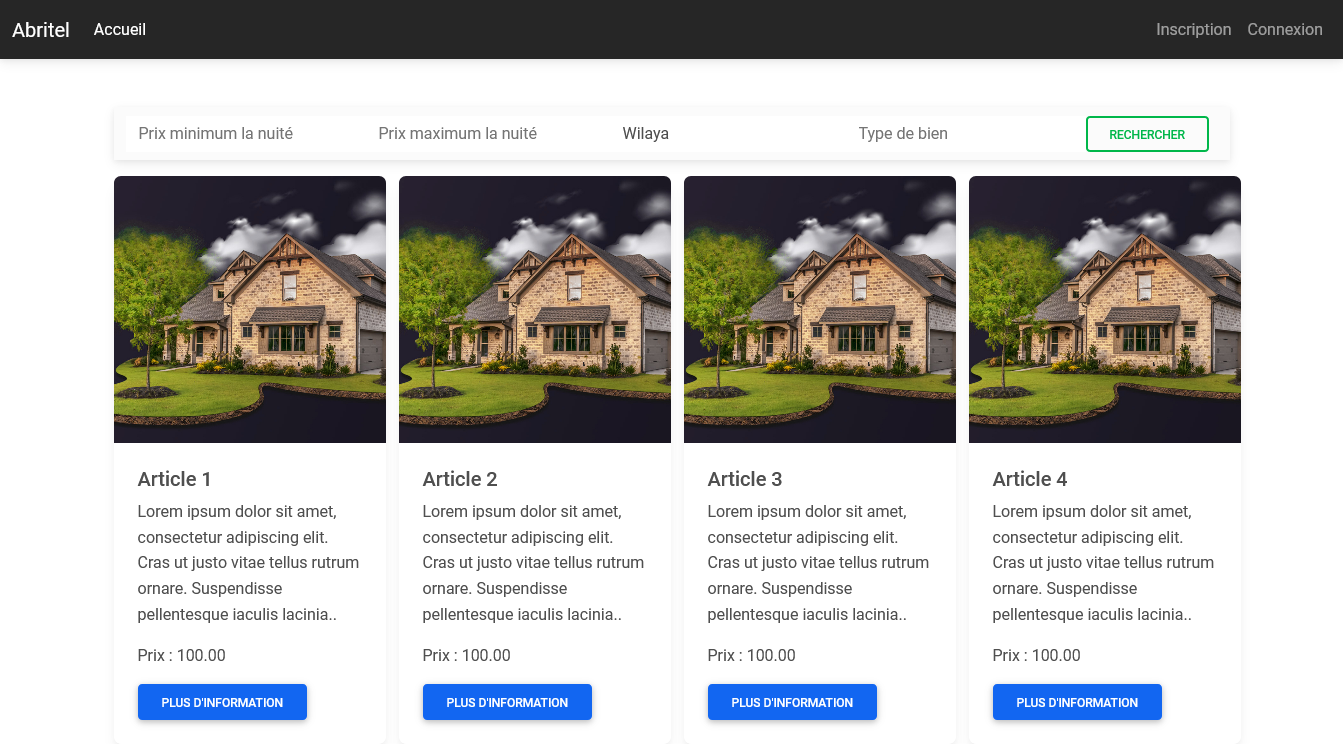
#### Ecran d’accueil :

Contient une barre de navigation et une barre de recherche centrale



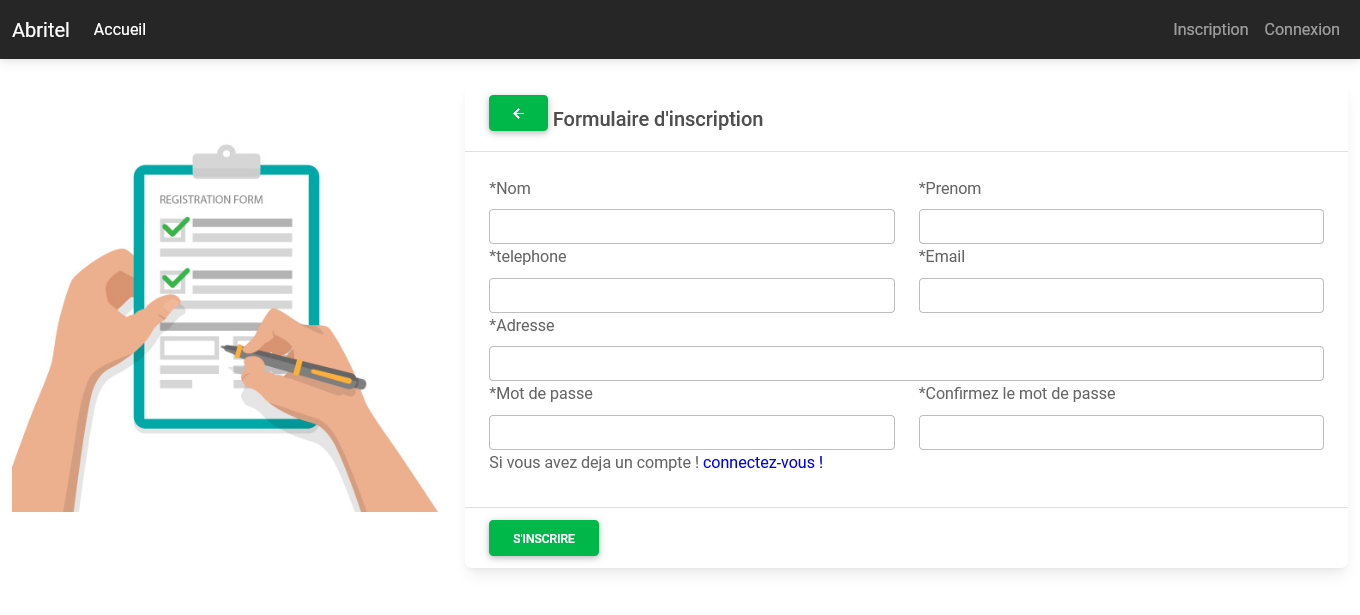
#### Page de recherche :

Page de recherche et filtre, contient une barre de filtre et affichage des annonces en cartes

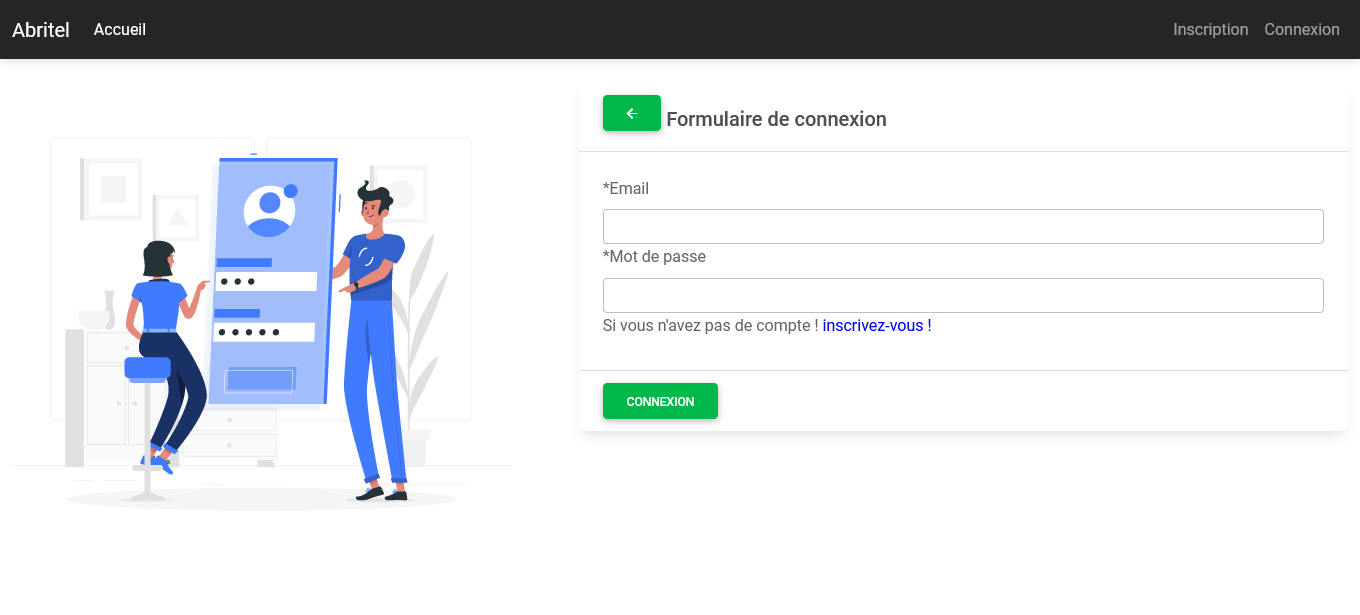


#### Page d’inscription :

Inscription universelle, avec proposition de connexion en cas de possession de compte

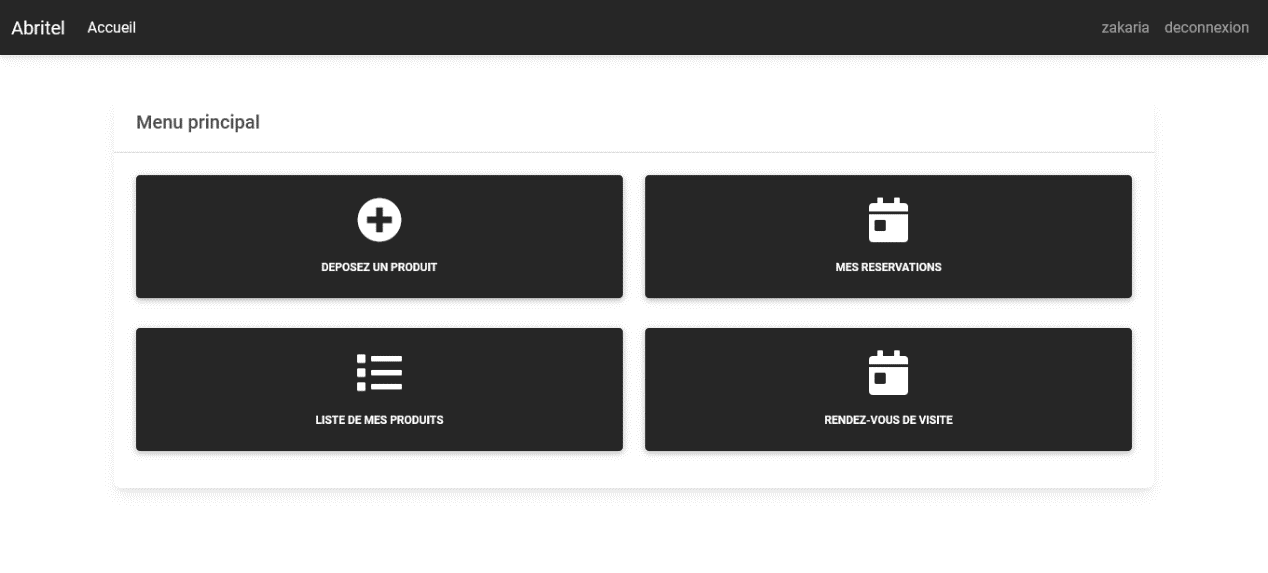


#### Page de connexion :

Page de connexion avec proposition de création de compte en cas de non existence de compte

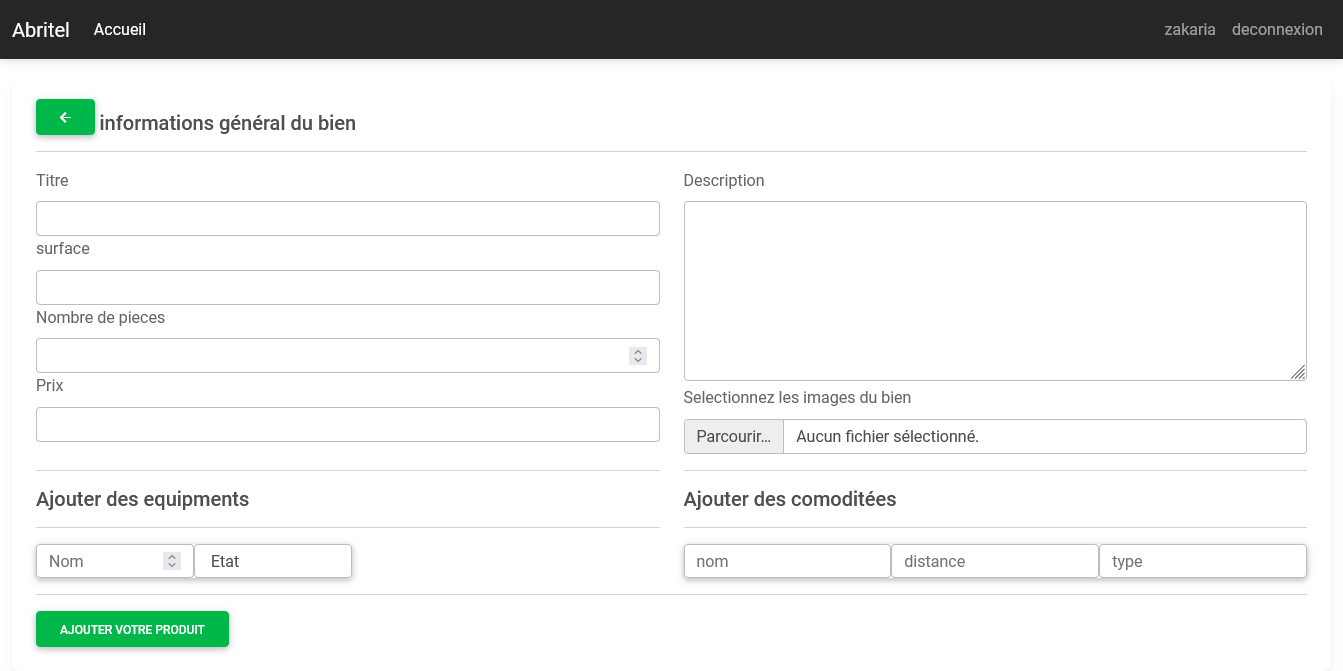
Page home :

Affiche le menu standard dédier a l’utilisateur



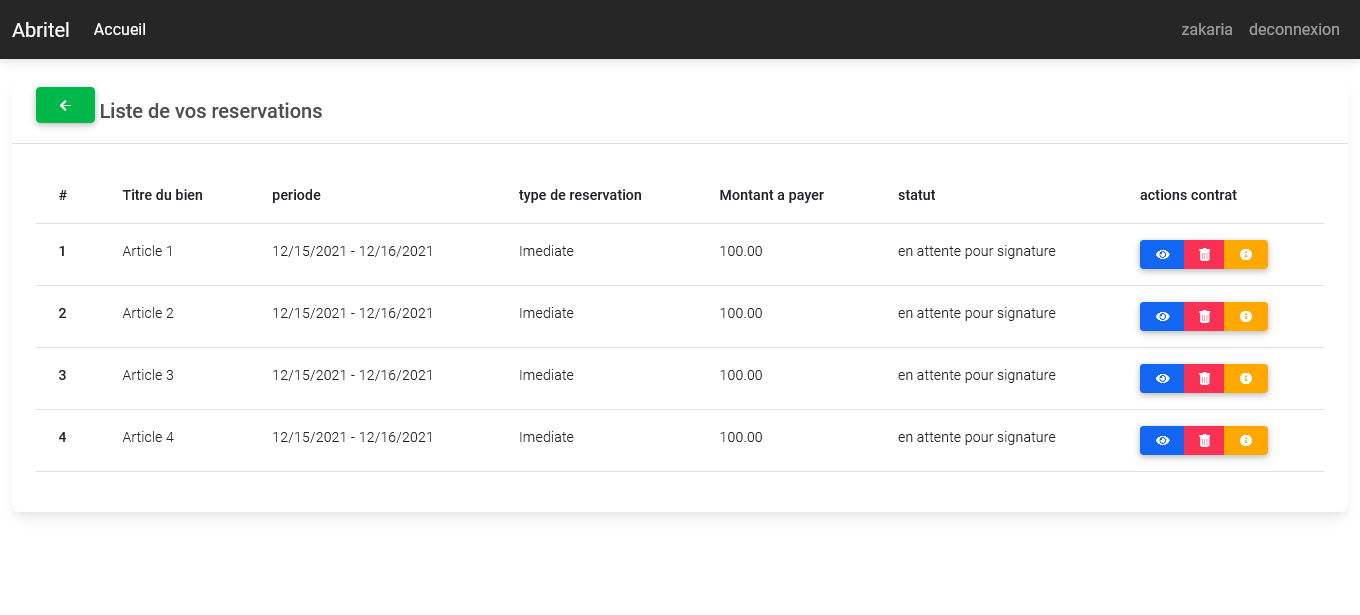
#### Page ajout de produit :

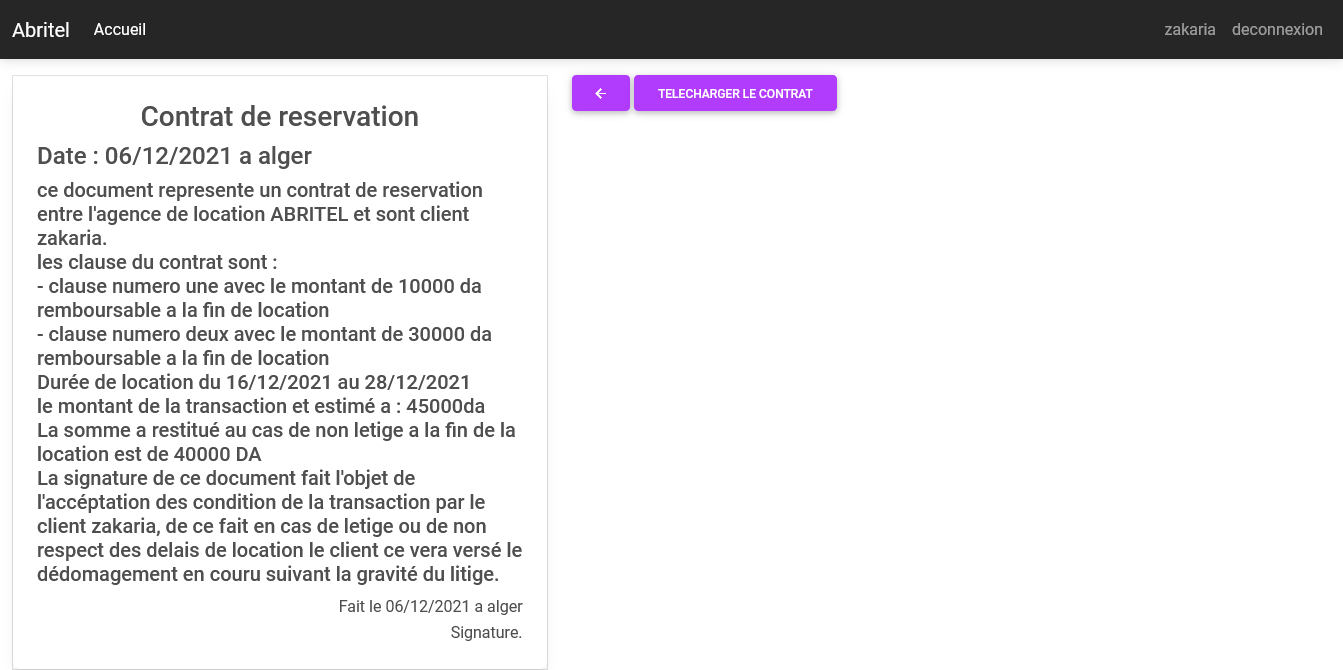
Affiche un formulaire d’ajout de produit (bien)

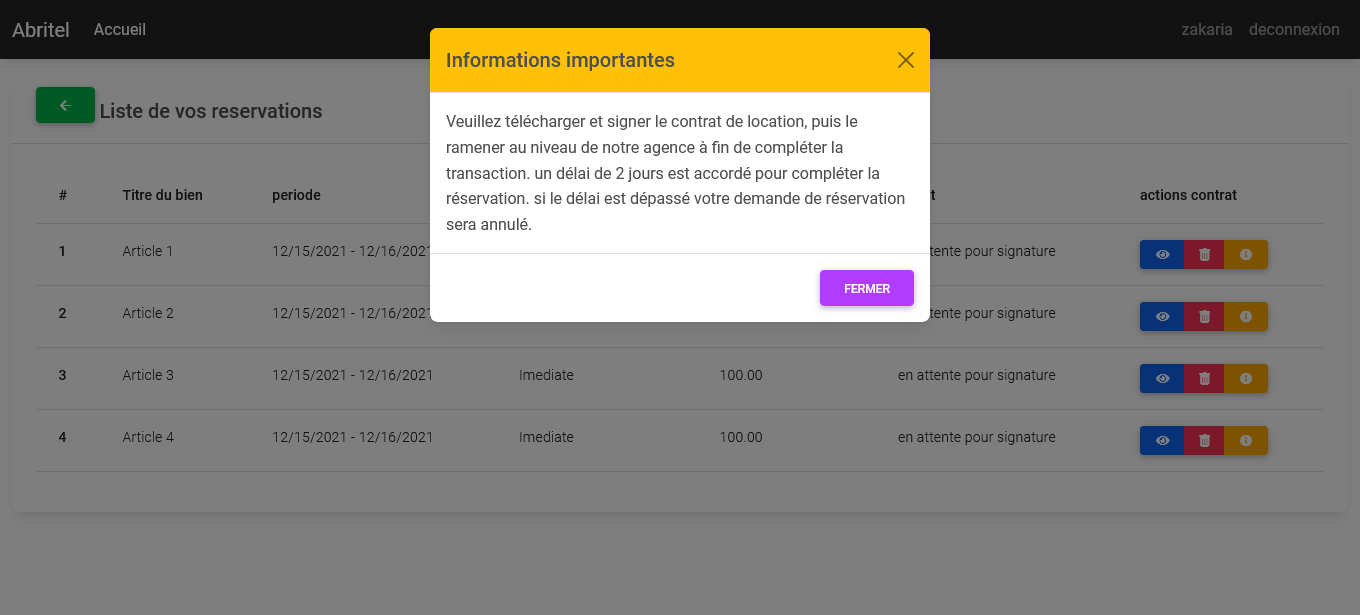


#### Page liste des réservations :

Affiche un tableau des réservations avec interaction :

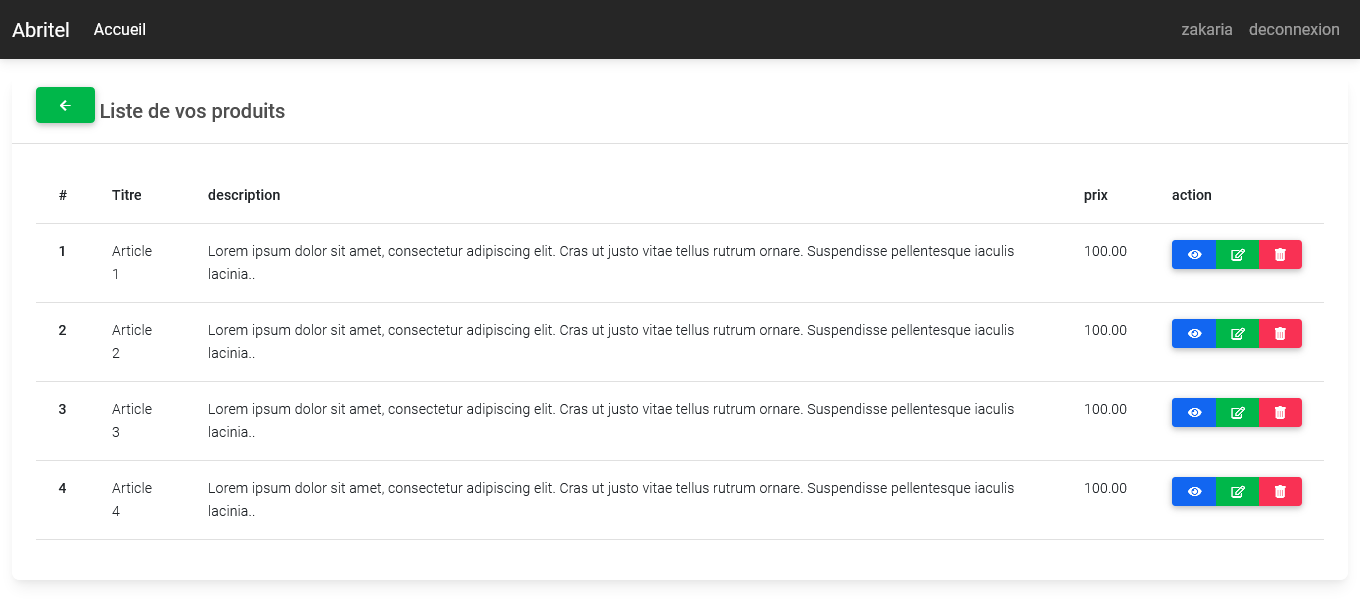


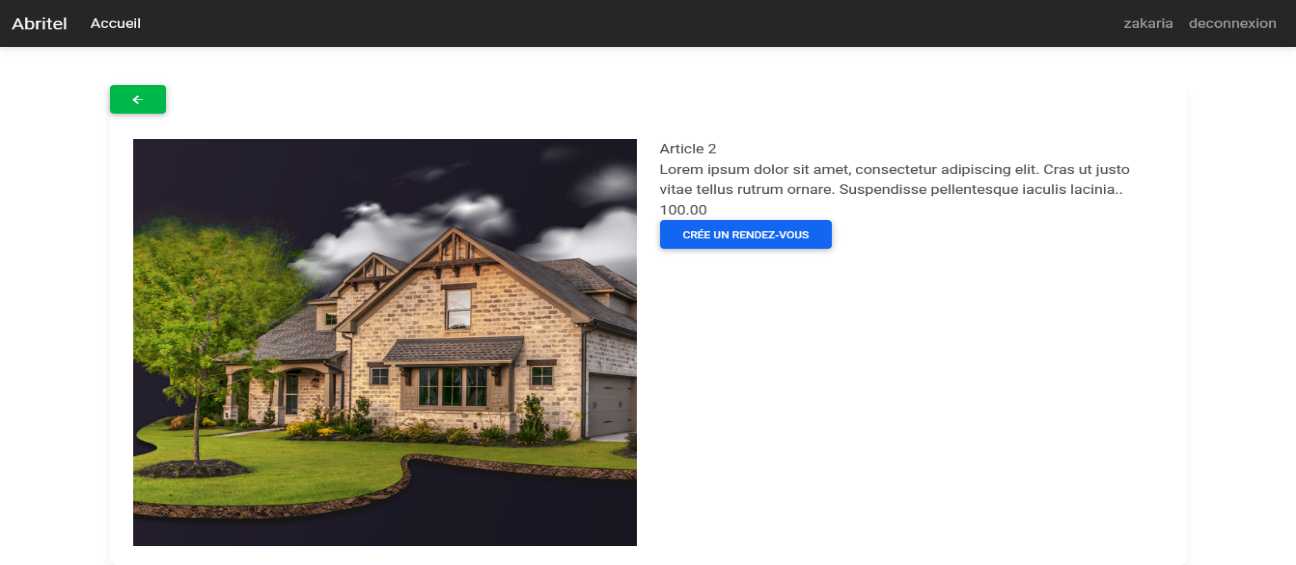




#### Page liste des produits :

Affiche un tableau des annonces de produits avec interaction :





#### Page liste des rendez-vous :

Affiche un tableau des rendez-vous avec interaction :