

Déploiement

Gestionnaire de Réservation de Salles GRS

Mariam BOUHAJJA

StreamIO

date de création : 22/04/2024

dernière date de modification: 08/06/2024

Sommaire :

Sommaire :.....	2
Historique de modification :.....	2
Déploiement du Frontend.....	3
Déploiement de l'API et de la base de données.....	5

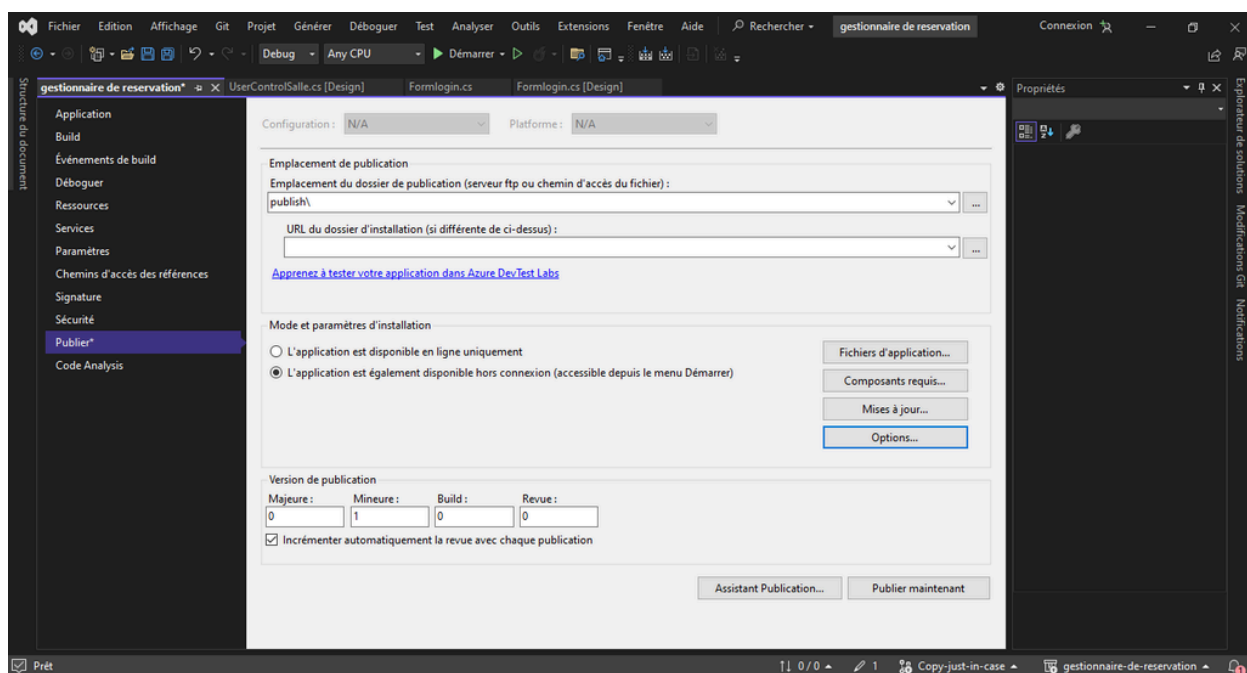
Historique de modification :

Réviseur	Date	Description
Mariem Bouhajja	22/04/2024	Création du document
Mariem Bouhajja	08/06/2024	Relecture et correction de l'orthographe

Déploiement du Frontend

Visual Studio 2022 permet un déploiement facile et intuitive du frontend

Il suffit d'aller dans **Explorateur de solutions > Paramètres > Publier**. Une fois les informations de l'application renseignées, j'ai généré un certificat pour garantir l'intégrité de l'application. Le fichier exécutable sera alors créé.



Options de publication

Description
Déploiement
Manifestes
Associations de fichiers

Langue de publication : Français

Nom de l'éditeur : Mariem

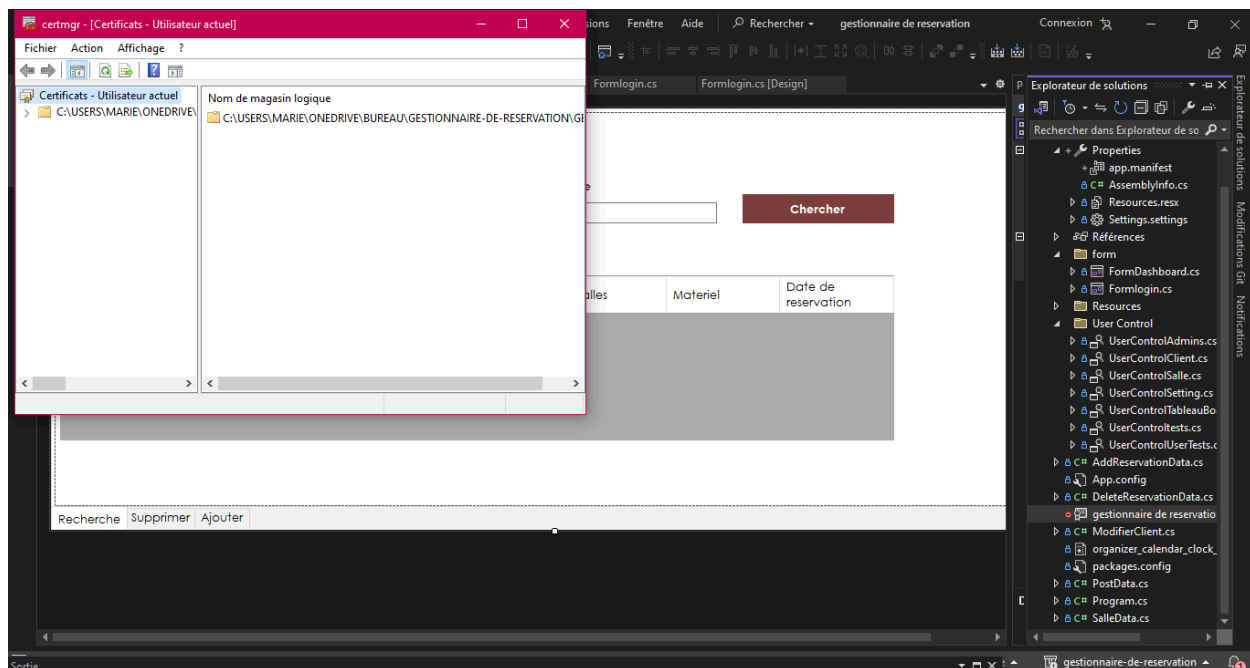
Nom de la suite : Applications de réservation

Nom du produit : Gestionnaire de Réservation

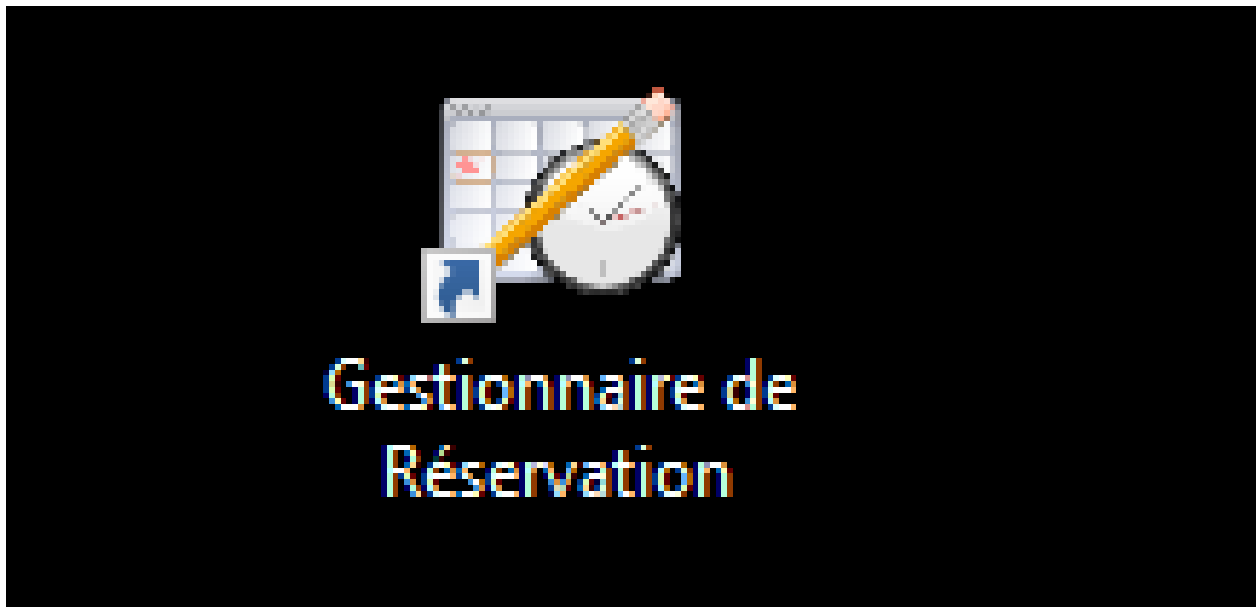
URL du support technique :

URL d'erreur :

OK Annuler



génération du certificat



le fichier exécutable est créé

Déploiement de l'API et de la base de données

dépendances installées pour le déploiement :

- git => pour pouvoir installer les fichiers du code
- nodejs et npm pour faire tourner le serveur de l'api
- maria-db (en installation secure) serveur pour la gestion de la base de donnée

```

lecture des informations d'état... Fait
:: Impossible de trouver le paquet apahe2
sioslam@sioslam:~$ sudo apt install git nodejs npm mariadb-server apache2
lecture des listes de paquets... Fait
construction de l'arbre des dépendances... Fait
lecture des informations d'état... Fait
les paquets suivants ont été installés automatiquement et ne sont plus nécessaires :
  libflashrom1 libftdi1-2 libllvm13
Veuillez utiliser « sudo apt autoremove » pour les supprimer.
les paquets supplémentaires suivants seront installés :
  apache2-bin apache2-data apache2-utils binutils binutils-common binutils-x86-64-linux-gnu build-essen
  gawk gcc gcc-11 git-man gyp javascript-common libalgorithm-diff-perl libalgorithm-diff-xs-perl libalg
  libaprutil1-dbd-sqlite3 libaprutil1-ldap libasan6 libbinutils libc-ares2 libc-dev-bin libc-devtools l
  libcgi-pm-perl libconfig-inifiles-perl libcrypt-dev libctf-nobfd0 libctf0 libdaxctl1 libdbd-mysql-per
  libfakeroot libfcgi-bin libfcgi-perl libfcgi0ldbl libfile-fcntllock-perl libgcc-11-dev libhtml-templa
  libjs-highlight.js libjs-inherits libjs-is-typedarray libjs-psl libjs-source-map libjs-sprintf-js lib

```

Traitement des actions différées (« triggers ») pour libc-bin (2.35-0ubuntu3.6) ...

```

sioslam@sioslam:~$ sudo mysql_secure_installation

NOTE: RUNNING ALL PARTS OF THIS SCRIPT IS RECOMMENDED FOR ALL MariaDB
      SERVERS IN PRODUCTION USE!  PLEASE READ EACH STEP CAREFULLY!

In order to log into MariaDB to secure it, we'll need the current
password for the root user. If you've just installed MariaDB, and
haven't set the root password yet, you should just press enter here.

Enter current password for root (enter for none):
OK, successfully used password, moving on...

Setting the root password or using the unix_socket ensures that nobody
can log into the MariaDB root user without the proper authorisation.

You already have your root account protected, so you can safely answer 'n'.

Switch to unix_socket authentication [Y/n] ☐

```

une fois les dépendances installés il faut copier le projet présent dans un dépôt git avec la commande "git clone"

```
sioslam@sioslam:/opt/mariam/application-de-reservation/backend$ ll
total 108
drwxr-xr-x  3 root root  4096 avril 22 15:01 ./
drwxr-xr-x  5 root root  4096 avril 22 14:36 ../
-rw-r--r--  1 root root  2900 avril 22 14:36 app.test.js
-rw-r--r--  1 root root   699 avril 22 15:01 database.js
-rw-r--r--  1 root root    12 avril 22 14:36 .gitignore
-rw-r--r--  1 root root 15486 avril 22 14:36 index.js
drwxr-xr-x 150 root root  4096 avril 22 14:54 node_modules/
-rw-r--r--  1 root root   360 avril 22 14:53 package.json
-rw-r--r--  1 root root 61955 avril 22 14:54 package-lock.json
sioslam@sioslam:/opt/mariam/application-de-reservation/backend$
```

Déploiement de la base de données :

j'ai mis en place la base de données avec la commande "sudo mysql -u -p < mabasededonne.sql"

```
near Database at line 1
MariaDB [(none)]> show databases;
+-----+
| Database |
+-----+
| application_de_reservation |
| information_schema |
| mysql |
| performance_schema |
| sys |
+-----+
5 rows in set (0,001 sec)

MariaDB [(none)]>
```

il faut après créer un utilisateur et lui donner les bon droits sur la bdd

```
MariaDB [(none)]> Create USER mariem@localhost IDENTIFIED BY '';
```

```
MariaDB [(none)]> GRANT ALL PRIVILEGES ON application_de_reservation.* TO mariem@localhost;
```

```

near USER at line 1
MariaDB [(none)]>
MariaDB [(none)]> Select user from mysql.user;
+-----+
| User          |
+-----+
| mariadb.sys   |
| mariem        |
| mysql         |
| root          |
+-----+
4 rows in set (0,105 sec)

MariaDB [(none)]> 

```

Déploiement d'un service qui lance automatiquement l'API:

Pour créer le script j'ai utilisé la commande "`_etc_/systemd/system/serveurNodeMariem.service`" et j'ai écrit le script suivant

```

GNU nano 6.2 /etc/systemd/system/serveurNodeMariem.service
[Unit]
Description=My Node.js App
After=network.target

[Service]
User=sloslam
WorkingDirectory=/opt/mariem/application-de-reservation/backend
ExecStart=node /opt/mariem/application-de-reservation/backend/index.js
Restart=always
RestartSec=10

[Install]
WantedBy=multi-user.target

```

GNU nano 6.2 /etc/systemd/system/serveurNodeMariem.service
 [Unit]
 Description=My Node.js App
 After=network.target
 [Service]
 User=sloslam
 WorkingDirectory=/opt/mariem/application-de-reservation/backend
 ExecStart=node /opt/mariem/application-de-reservation/backend/index.js
 Restart=always
 RestartSec=10
 [Install]
 WantedBy=multi-user.target

Alt Aide Alt Écrire Alt Chercher Alt Couper Alt Exécuter Alt Enplacement Alt Annuler Alt Marquer
 Ctrl Quitter Ctrl Lire fich. Ctrl Remplacer Ctrl Coller Ctrl Justifier Ctrl Aller ligne Ctrl Refaire Ctrl Copier Ctrl -> Crochet Ctrl Retrouver

Pour activer le service j'ai utilisé les commandes suivantes :

```
sudo systemctl daemon-reload
```

```
sudo systemctl enable serveurNode.service
```

```
sudo systemctl start serveurNode.service
```

```
sioslam@sioslam:~$ sudo systemctl status serveurNodeMariem.service
[sudo] Mot de passe de sioslam :
● serveurNodeMariem.service - My Node.js App
   Loaded: loaded (/etc/systemd/system/serveurNodeMariem.service; enabled; vendor preset: enabled)
   Active: active (running) since Thu 2024-04-25 15:58:17 CEST; 5min ago
     Main PID: 25731 (node)
       Tasks: 11 (limit: 4452)
      Memory: 14.5M
         CPU: 374ms
    CGroup: /system.slice/serveurNodeMariem.service
            └─25731 node /opt/mariem/application-de-reservation/backend/index.js

avril 25 15:58:17 sioslam systemd[1]: Started My Node.js App.
avril 25 15:58:17 sioslam node[25731]: le serveur tourne!
sioslam@sioslam:~$
```

Finalement, il suffit de changer les adress ip dans l'API par l'adress IP fixe générée par les administrateurs système réseaux

```
4 références
public partial class UserControlUserTests : UserControl
{
    private readonly string apiUrl = "http://172.20.90.7/afficherReservation";
    private int selectedID;
    private string selectedDate;
```