

Documentation technique pour le logiciel **RFID Access pour LE LAB**



Développeur et rédacteur : Mattéo Volpi

Sommaire :

I. Composition du dossier « rfid_access »

II. Plus de détails sur la base de donnée

Annexe : Mode d'emploi

I. Composition du dossier « rfid_access »

Le dossier rfid_access contient :

- Trois fichiers exécutables
- Les fichiers contenant le code source
- Un dossier ressources
- Un dossier arduino
- Un dossier spécifique à python
- Un fichier « requirements.txt »

Les fichiers exécutables :

Le fichier principal : « rfid_access.py »

Il démarre l'application et doit être lancé en ligne commande, comme ceci : **./rfid_access.py**

Le fichier : « clear_db.py »

C'est un programme qui sert à vider les logs et les statistiques annuelles de base de donnée. Peut aussi servir à supprimer les utilisateurs enregistré.

Le fichier : « find_user.py »

Ce programme sert de chercher des informations sur un les entrées et sorties d'un utilisateur en particulier ou bien d'une journée en particulier.

Les fichiers sources :

Le nombre de fichier source s'élève au nombre de onze (les fichiers exécutables sont aussi des fichiers sources).

Le dossier ressources :

Le dossier ressource contient les ressources graphiques dont se sert le logiciel, ne pas y toucher sous peine d'empêcher le logiciel de démarrer.

Le dossier arduino :

Le dossier arduino contient le code utilisé dans l'arduino pour capter les IDs des cartes et faire rugir l'enceinte.

Le dossier spécifique à python :

Le dossier se nomme “__pycache__” c'est un dossier que le langage python utilise pour stocker les versions compilées du code. Il ne craint pas d'être supprimé, dans le doute, ne pas y toucher.

Un fichier “requirements.txt” :

Le fichier requirements.txt liste les dépendances du logiciel.

Détails supplémentaires sur les fichiers sources :

Admin.py : Contient la page d'administration dans le logiciel, permet d'exporter les données en CSV, de changer le mot de passe, de supprimer la liste des inscrits ...

Database.py : Contient toutes les méthodes permettant d'interagir avec la base de donnée.

font.py : Contient une variable représentant la police d'écriture.

id_call.py : Contient un objet global pour accéder à la base de donnée partout dans le code.

List.py : Contient la page listant les utilisateurs présents au lab, permet d'ajouter ou supprimer manuellement des utilisateur de la liste ...

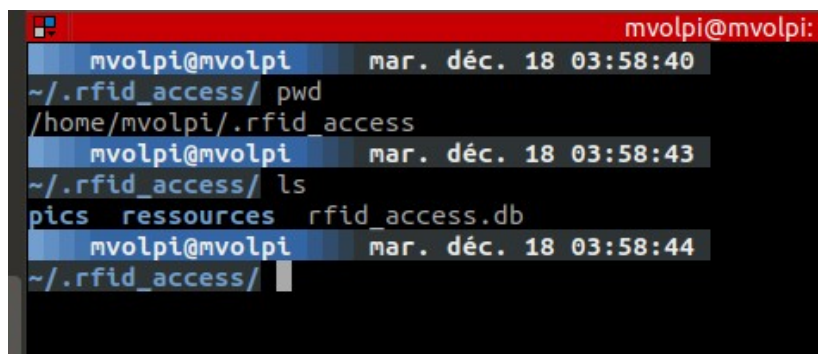
Page.py : Contient l'objet Page dont toutes les pages héritent (Admin, List, Register et Stats).

Register.py : Contient la page pour enregistrer un utilisateur, nécessite le nom, le prénom, l'âge, la classe, le lieu de résidence et une photo pour enregistrerl'utilisateur correctement.

Stats.py : Contient la page où s'afficher les statistiques, permet de visualier différentes statistiques.

II. Détails sur la base de donnée

La base de donnée est située dans un dossier dans le home :



```
mvolpi@mvolpi: ~  
mar. déc. 18 03:58:40  
~/rfid_access/ pwd  
/home/mvolpi/.rfid_access  
mvolpi@mvolpi: ~  
mar. déc. 18 03:58:43  
~/rfid_access/ ls  
pics  ressources  rfid_access.db  
mvolpi@mvolpi: ~  
mar. déc. 18 03:58:44  
~/rfid_access/
```

On peut voir grâce à la commande “pwd” que l’on se situe dans le dossier “.rfid_access”, lui-même situé dans le home de notre PC (ici “/home/mvolpi”). On voit grâce à la commande “ls” qu’un fichier se nommant “rfid_access.db” se trouve dans ce dossier, c’est notre base de donnée.

Comment est organisée la base de donnée :

Pour la consulter vous pouvez taper la commande suivante :
“sqlitebrowser ~/.rfid_access/rfid_access.db &”

Une fenêtre va s’ouvrir. On peut y voir plusieurs onglets, ceux qui nous intéressent sont les onglets « Database Structure » et « Browse Data ». Dans l’onglet « Database Structure » on retrouve les composants des différentes tables et dans l’onglet « Browse Data » on retrouve le contenu des différentes tables.

Tables :

annual : Contient les statistiques journalières, une ligne correspond à une journée et les cases de cette ligne correspondent aux données de la journée.

daily : Contient les entrées et les sorties journalières des utilisateurs, cette table est vidée chaque jour.

log : Contient toutes les entrées et sorties des utilisateurs depuis la mise en marche du logiciel. Cela permet de savoir qui était présent au Lab au moment d’un accident par exemple.

sql_sequence : Table générées par sql, contient les id des lignes de la table annual, ne pas toucher.

users : Contient tout les utilisateurs enregistré, id de la carte, nom, prénom, âge, classe, commune de résidence, lien vers la photo.