

Dossier personnel

Doumenc Thomas

Lycée Victor Hugo 31770 Colomiers

Projet BUBO 2020



BUBO

Sommaire

Sommaire	2
Introduction au projet	3
Contexte:	3
Enjeux:	3
Objectif:	4
Technologies utilisées	4
Oracle VM VirtualBox:	4
Fedora 24/31:	5
NetBeans:	5
Google Drive:	5
Time Performance:	5
Sqlite3:	5
C++:	5
HTML5/javascript:	5
Problème format d'altitude	6
Présentation de l'existant	6
Recherche du problème	7
Solutions misent en place	9
Test de mise en oeuvre	10
Problèmes rencontrés	11
Téléchargement BDD format texte	11
Présentation de l'existant	12
Recherche de solutions	13
Test de mise en oeuvre	16
Suivi d'activité	18
Bilan	20

Introduction au projet

Contexte:

Nous avons pris le relais sur le projet BUBO confié par DarkSkyLab, une entreprise qui se définit comme un bureau d'étude spécialisé dans l'expertise scientifique de la pollution lumineuse et de ses impacts sur les écosystèmes.

DarkSkyLab a pour mission la réalisation de diagnostics d'éclairage public et privés pour en optimiser l'usage afin de protéger l'environnement nocturne et réaliser des économies sur les dépenses énergétiques.

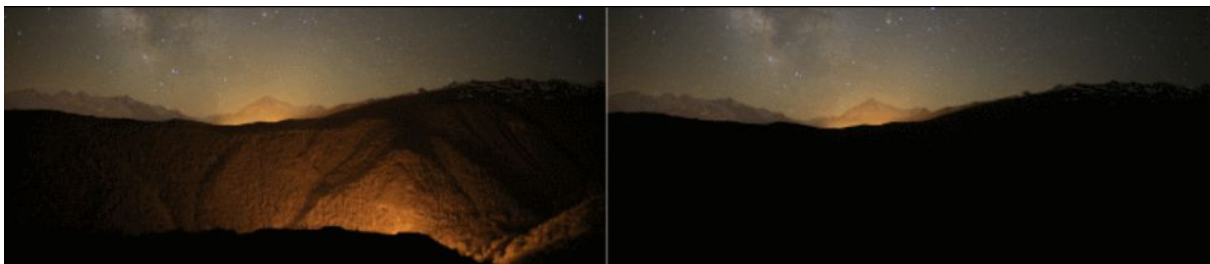


Enjeux:

Cette mission plusieurs enjeux centré autour de la protection de l'environnement nocturne:

- Optimisation des éclairage afin de réaliser des économies énergétique.
- Réduire l'impact de la pollution lumineuse sur la biodiversité et la santé publique.
- Rétablir le lien entre les sociétés et leur environnement nocturne

L'exemple de Aulon:



Objectif:

BUBO est une solution embarquée sur Raspberry Pi permettant d'assister l'opérateur de terrain lors de la collecte de données utiles au diagnostic d'un site. BUBO utilise une interface web mobile.

Les objectifs du groupe cette année par ordre de priorité:

- 1- Champ Altitude sous forme d'un entier et pas d'un BLOB
- 2- Maintien du contexte (valeurs de champs) en cas de rechargement de la page
- 3- Mettre à jour la date et l'heure du Raspberry Pi dès qu'un fix est réalisé et ce de manière systématique.
- 4- Ajouter une option dans la page des préférences qui permet de déclencher une géolocalisation dès que l'on appuie sur le bouton "Nouveau".
- 5- Permettre le paramétrage des informations d'envoi des données par courriel dans la page des options
- 6- Permettre un export au format textuel pour sauver un fichier avec tous les enregistrements dans le téléphone
- 7- Ajouter l'heure et la date de l'enregistrement dans la base de données
- 8- Gestion des photos

Lors de la première répartition des tâches j'ai été assigné aux numéros 1, 3 et 6.

Tout au long du projet nous gardon contacte avec DarkSkyLab au travers de M. Deverchère.

Technologies utilisées

Oracle VM VirtualBox:

Compte tenu des événements de cette fin d'année j'ai eu besoin de travailler majoritairement à domicile, Oracle VM VirtualBox à été



Doumenc Thomas

pour moi la meilleure solution afin de retrouver un environnement de travail similaire a celui que j'avais au lycée.

Fedora 24/31:

Fedora est l'environnement avec lequel je suis le plus à l'aise notamment quand il s'agit de travailler en parallèle avec une raspberry. C'est donc naturellement vers lui que je me suis tourné dès le départ du projet. J'ai par la suite installé une VM avec la version 31 à mon domicile.



NetBeans:

C'est l'IDE qui ma permis de coder et naviguer entre les fichiers du projet facilement.



Google Drive:

Un cloud web grâce auquel mon groupe a pu facilement mettre en commun toutes les ressources dont nous avons besoin.



Time Performance:

C'est un logiciel de gestion de projets qui nous a permis d'organiser le projet de façon optimale notamment via un système de tableau de tâche.



Sqlite3:

Sqlite est une librairie en langage C qui implémente un moteur de base de données.

C++:

C'est le langage qui a été choisi pour le projet.



HTML5/javascript:

Le système passe par une interface web mobile faite dans ces langages.



Problème format d'altitude

Présentation de l'existant

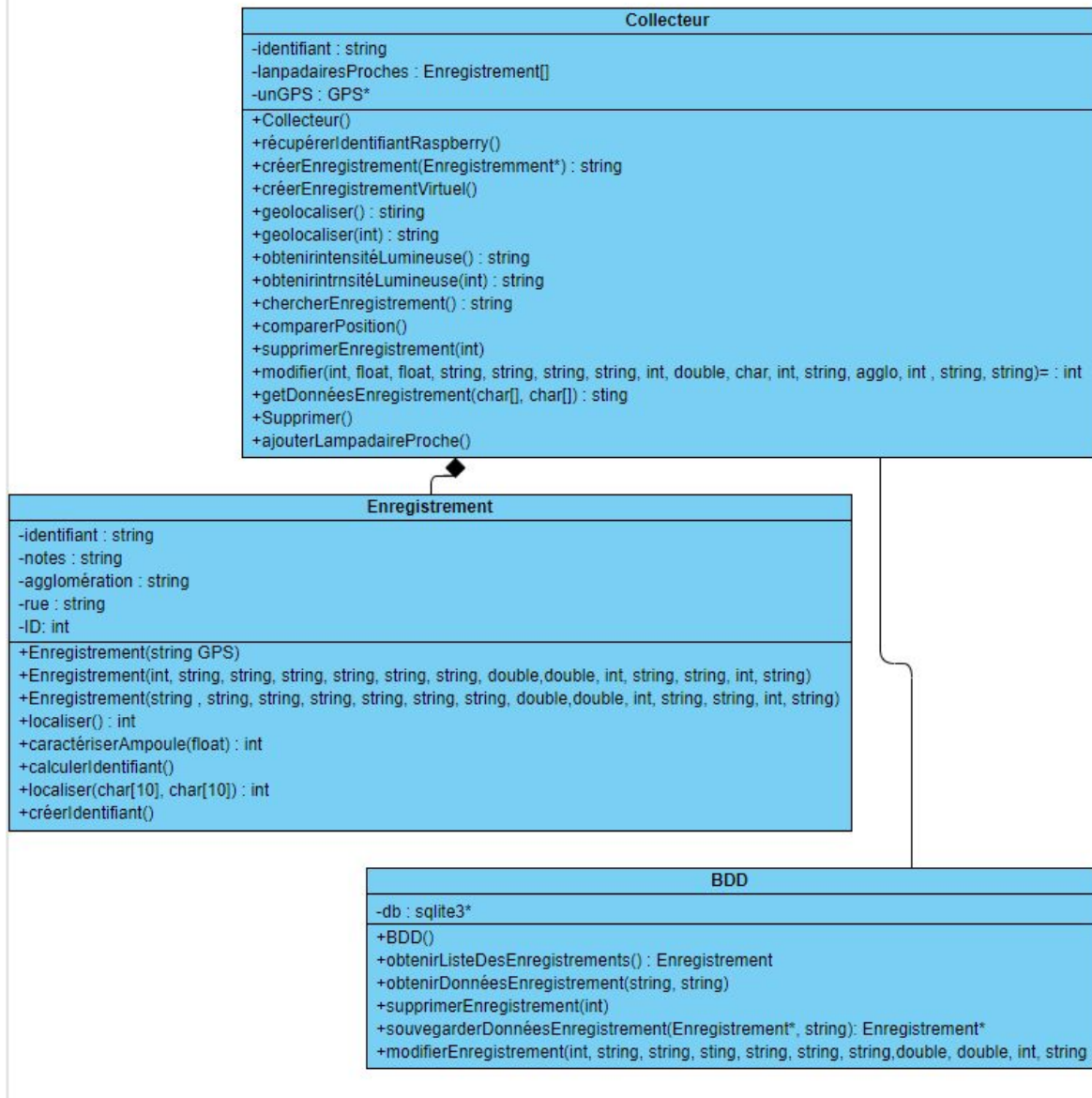
Pour travailler sur ce problème j'ai eu à utiliser plusieurs éléments, le premier est l'interface web mobile qui m'a servie à faire des essais et avoir une première vue du souci avec le tableau en première page qui résume les derniers enregistrements effectués.

Le deuxième élément c'est la base de données, mais surtout DB Browser for SQLite qui m'a offert une interface simple pour analyser le contenu de la base de données Enregistrement où est localisée l'Altitude.

Enfin j'ai travaillé sur la partie programmation objet du projet et notamment les classes BDD et Enregistrement.

The screenshot shows the Bubo web interface. At the top, there's a logo 'Bubo' and an owl icon. Below the logo are two buttons: 'Rafrachir' and 'Revue'. The main part of the interface is a table with four columns: ID, LAT., LONG., and ULOR. The table contains several rows of data, with the row having ID 9 highlighted. Below the table are three buttons: 'Nouveau', 'Supprimer', and 'Modifier'. Further down are two buttons: 'Geolocaliser' and 'Enregistrer'. Below these are two input fields: 'Latitude:' and 'Longitude:'. The 'Latitude:' field contains the value '43.615828' and the 'Longitude:' field contains '1.309400'. Below these are two more input fields: 'Type de Lampadaire:' and 'ULOR:'. The 'Type de Lampadaire:' field has a dropdown menu showing 'Applique projecteur' and the 'ULOR:' field has a dropdown menu showing '100%'.

ID	LAT.	LONG.	ULOR
17	43.518821	1.312915	30
16	43.518813	1.312883	0
13	43.518920	1.312433	0
10	43.615808	1.309400	30
9	43.615828	1.309400	100
6	43.614473	1.305433	25
4	43.614573	1.305433	20
3	43.518756	1.312950	15



Recherche du problème

Tout d'abord j'ai pu voir dans le tableau sur la page web que l'altitude s'affichait sous forme de caractères cabalistiques tel que "Ã©". DB Browser ne pouvant pas lire ces caractères il affichait alors "BLOB".

Ensuite après avoir étudié les éléments en liens avec le champ altitude, je me suis rendu compte que c'était un gros bazar au niveau des types de variable où le contenu du champ était récupéré dans un tableau de 4 char et transformé en int lors de la géolocalisation.


```
141 void Geolocalisation::calculAltitude(string leChamps[200])
142 {
143     char alti[4];
144     int intalti;
145     strcpy(alti, leChamps[9].c_str());
146     intalti = atoi(alti);
147     snprintf(altitude, 5, "%d", intalti);
148 }
149 }
```

Puis récupérer sous forme de tableau de 5 char au niveau d'Enregistrement.cpp

```
141 void Geolocalisation::calculAltitude(string leChamps[200])
142 {
143     char alti[4];
144     int intalti;
145     strcpy(alti, leChamps[9].c_str());
146     intalti = atoi(alti);
147     snprintf(altitude, 5, "%d", intalti);
148 }
149 }
```

Pour être finalement transformé en string dans BDD.cpp

```
179     string laAltitude=string(altitude, 5);
```

Hypothèse 1: A ce moment là j'ai donc pensé qu'à un moment lors du passage d'un tableau de 4 char à 5 char on récupérerait un caractère comme une lettre dans la mémoire de la PI. Ou alors qu'à un moment, on perdait le caractère nul de fin de string.

Par la suite, j'ai pensé que le problème survenait lors du dernier passage de tableau de char à string (l.179 de BDD.cpp)


```
173     string laLongitude=string(longitude);
174
175
176     string laLatitude=string(latitude);
177
178
179     string laAltitude=string(altitude,5);
180
```

Hypothèse 2: Le dernier passage de type est un “copier coller” de ceux pour la latitude et la longitude et ne convient pas à l’altitude.

Solutions misent en place

La première solution que j’ai essayé fut d’homogénéiser les variables de tous les fichiers d’abord à un tableau de 5 char puis à un tableau de 4 char. Mais suite au test je me suis rendu compte que ça n’avait eu aucun effet.

C’est à ce moment là que j’ai commencé à m’intéresser au lignes 179 et à la 209 de BDD.cpp.

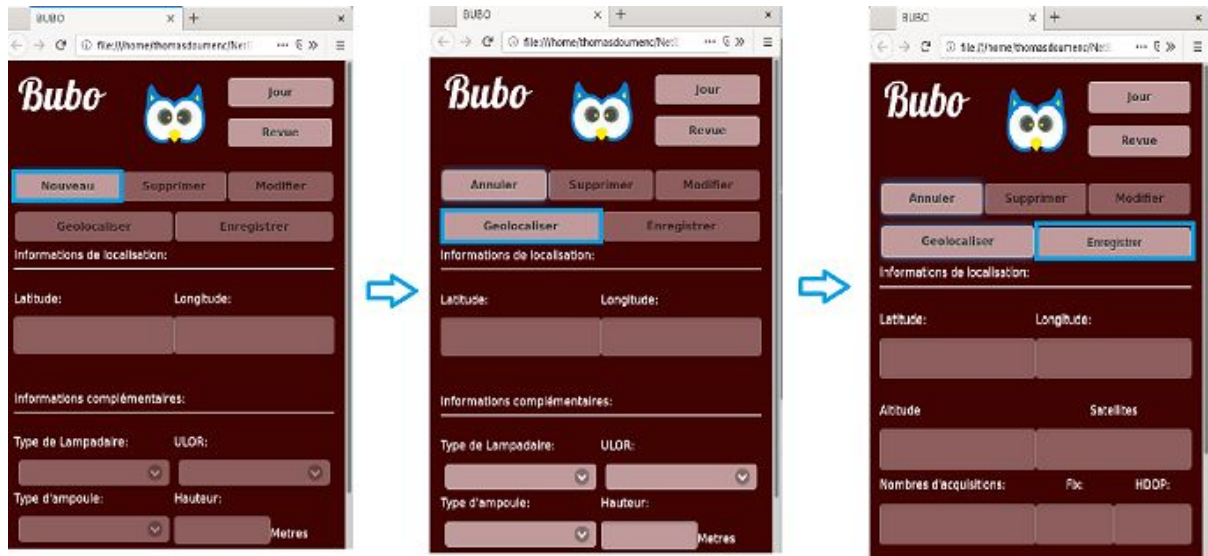
```
209     sqlite3_bind_text(stmt, 10, laAltitude.c_str(), laAltitude.length(), 0);
```

Je me suis rendu compte que l’altitude devait être traitée comme la variable ULOR c’est à dire comme un int étant donné que l’altitude n’ira jamais en dessous de 0 et au dessus de 9999 pas besoin de la transformer en chaîne de caractères. J’ai donc modifié les deux lignes accordement qui sont donc devenues:

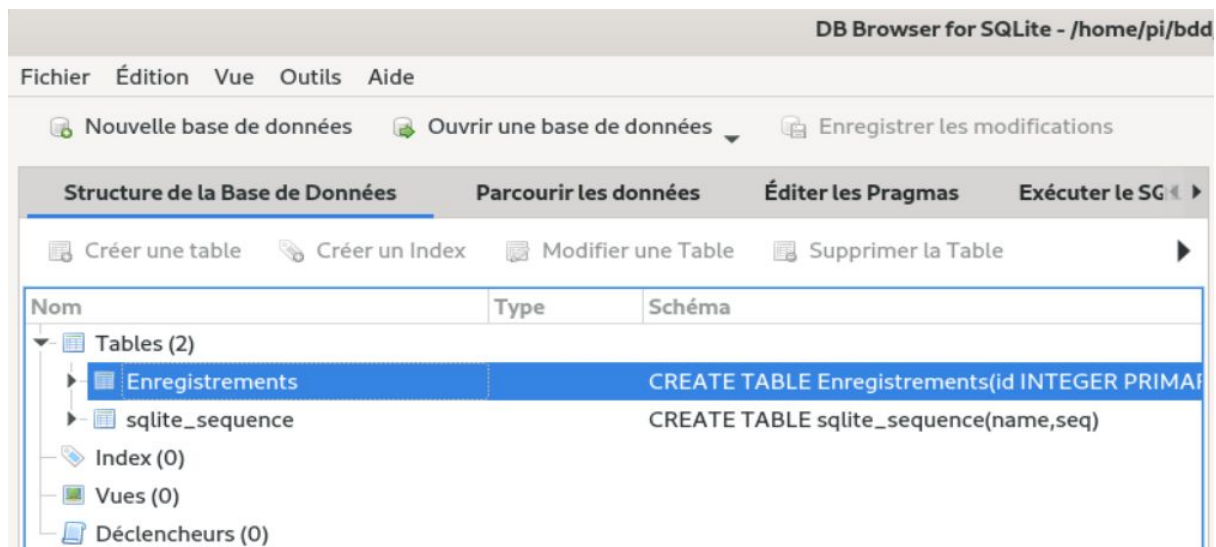
```
179     int laAltitude=stoi(altitude);
180
210     sqlite3_bind_int(stmt, 10, laAltitude);
211
```

Test de mise en oeuvre

Pour commencer ce test il nous a fallu faire un nouvel enregistrement.



Ensuite il a fallu transférer sur le poste de travail et ouvrir le fichier Enregistrement.db qui se trouve dans /home/pi/bdd avec le logiciel DB Browser for SQLite, puis sélectionner la table Enregistrements



Enfin on peut analyser les données. Comme on peut le voir sur la capture ci-dessous les deux enregistrements que j'ai réalisé suite à mise en place des corrections, le champ Altitude s'affiche maintenant dans un format correct.

Parcourir les données			Éditer le
			Nouvel Enregistrement
longitude	altitude	nbimages	
Filtre	Filtre	Filtre	
1.309468	BLOB	0	
1.30959	BLOB	0	
1.309533	181	0	
1.309533	181	0	

Problèmes rencontrés

Ne connaissant rien au cgi j'ai appris pas mal de choses en faisant des erreurs. Je pensais notamment que la compilation avait le chemin du dossier des cgi (/usr/local/apache2/cgi-bin) alors qu'il fallait envoyer le fichier dans son dossier manuellement

Téléchargement BDD format texte

Lors de la visite de Mr Deverchère pour nous expliquer ce qu'il attendait de notre groupe j'ai eu l'occasion de lui poser des questions au sujet de cette exportation. Les élèves des années précédentes étaient parti sur une solution d'envoi de la base de donnée par mail, Mr Deverchère souhaitant recevoir la base de données au format texte, j'avais fait des recherches préliminaire et trouver une solution utilisant openssh et une appli. Cependant ces solutions ne lui convenaient pas et il a alors parler d'un solution plus simple consistant en un bouton qui déclencherait le téléchargement du fichier en .txt.

Doumenc Thomas

Présentation de l'existant

Il existait une solution non fonctionnelle qui utilisait les cgi ainsi que l'architecture HTML5/JavaScript du projet ainsi le css.

Cette solution consistait en un bouton:



```
<div class="ui-block-b"><input type="button" id="envoibdd" value="Envoi bdd" ></div>
```

La partie JavaScript qui va avec

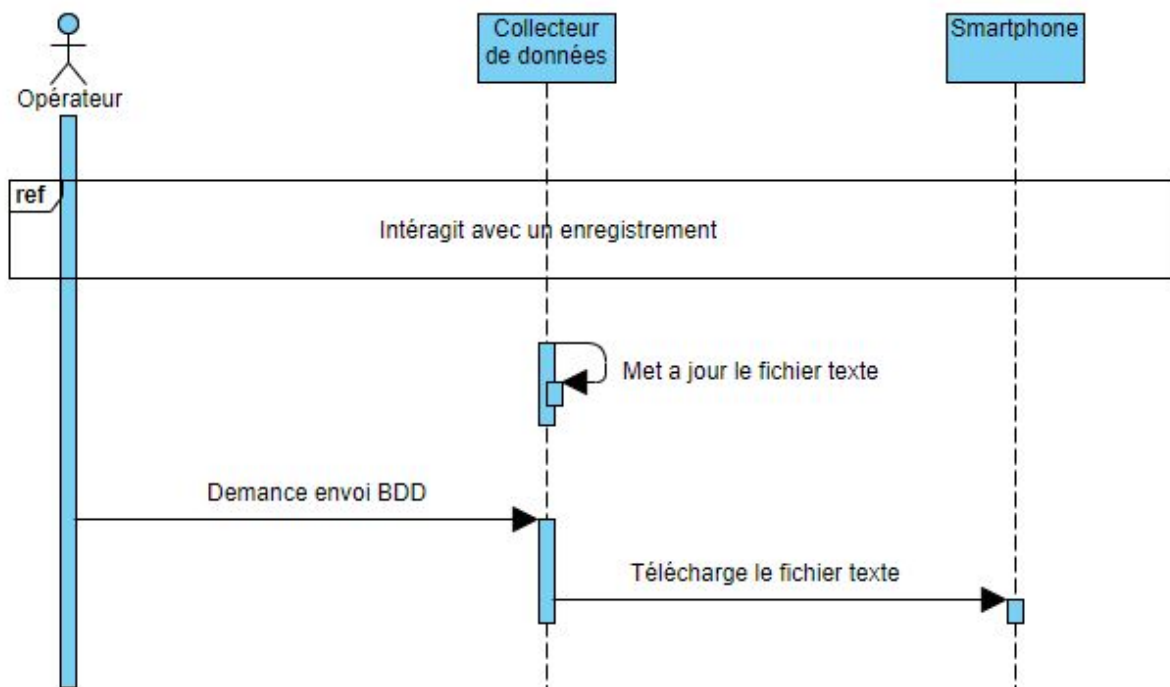
```
939 function envoimailbdd(){
940     var demandeConfirm = confirm("Voulez-vous transmettre la base de données ?");
941
942     if(demandeConfirm){ // bouton ok cliquer
943         $.ajax({
944             type: 'POST',
945             url: '../cgi-bin/envoyerBDD.cgi', //Lien du script cgi
946             dataType: 'json',
947
948             success: function (reponse,status) {
949                 switch (reponse.ERREUR)
950                 {
951                     case '1':
952                         alert('Envoi terminé. Vérifier la réception avant suppression.');
```

Et une cgi

```
20 int main()
21 {
22     cout << HTTPHTMLHeader();
23     Collecteur *leCollecteur= new Collecteur();
24     int testerreur=leCollecteur->envoyerbdd();
25     cout <<"{\n\t\"ERREUR\": \""+ to_string(testerreur)+"\"}\n\n";
26
27 }
```

Recherche de solutions

Après avoir reçus les attentes du client j'ai cherché une solution en essayant de faire simple et j'ai matérialiser l'idée que j'avais en tête sous forme de diagramme de séquence



J'ai donc commencé par chercher une solution pour récupérer le contenu de la table Enregistrements au format texte. Premièrement j'ai cherché une solution pour accéder au contenu de la table en ligne de commande, et j'y suis arrivé grâce au logiciel SQLite3. Il m'a permis d'afficher ceci:

```

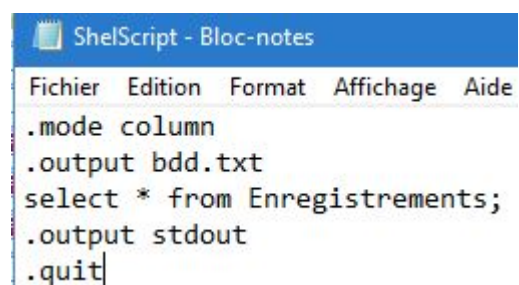
pi@bubo2:~/bdd $ sqlite3 Enregistrements.db
SQLite version 3.19.2 2017-05-25 16:50:27
Enter ".help" for usage hints.
sqlite> .headers on
sqlite> select * from Enregistrements;
id|power|albedo|flux|buboid|lamp|streetlight|height|latitude|longitude|altitude|
nbimages|town|street|ulor|date|notes
20|230|0.4|300.0|000000002ed09bce|Halogène|Projecteur sur mat|5.0|43.6182|1.3054
|100|0|Colomiers|rue des coquelicots|15|260519|pasdenote
21|230|0.4|300.0|000000002ed09bce|Halogène|Projecteur sur mat|5.0|43.6182|1.3054
|100|0|Colomiers|rue des coquelicots|15|260519|pasdenote
22|230|0.3|300.0|0000000020f09bce|Incandescence|Projecteur sur mat|5.0|43.614468
|1.3054|100|0|Colomiers|rue des coquelicots|15|250519|pasdenote
30|230|null|300.0|000000005952e188|Incandescence|Projecteur sur mat|5.0|43.51875
2|1.313025|161|0|Colomiers|rue des coquelicots|15|301019|pasdenote
33|0|||000000005952e188|||43.615897|1.309463|184|0|||0|030220|
34|0|||000000005952e188|||43.61586|1.309503|170|0|||0|030220|
35|0|||000000005952e188|||43.615902|1.309545|170|0|||0|030220|
36|0|||000000005952e188|Halogene|Candelabre de style|43.615702|1.30942|172|0
|||40|030220|

```

De là j'ai penser faire comme mon prédécesseur et reprendre le bouton en HTML5/Javascript qui ferait appel à un cgi. Cependant je ne suis pas arrivé à entrer des commandes dans l'interface sqlite via C++, c'est à dire qu'il m'était impossible de rentrer la commande "select * from Enregistrements;"

Je me suis ensuite penché plus en détail sur SQLite3 et sur un forum quelqu'un suggérait d'utiliser un script Shell. Le script Shell me permet de rentrer des commandes dans SQLite3 à partir du code C++. De plus il existe une commande SQLite3 qui permet de créer et mettre à jour un fichier texte avec le contenu d'une table.

Dans l'ordre ce script va changer l'affichage de la base de données en mode Colonne qui est d'après moi le plus lisible au format texte, puis créer le fichier bdd.txt, ensuite il va sélectionner l'entièreté de la table Enregistrements et la placer dans bdd.txt pour enfin quitter SQLite3.



```

ShelScript - Bloc-notes
Fichier  Edition  Format  Affichage  Aide
.mode column
.output bdd.txt
select * from Enregistrements;
.output stdout
.quit

```

Pour faire appel à la commande je voulais un endroit où la commande serait appeler les plus souvent possible, après la création d'un enregistrement, la modification d'un autre etc...

La commande étant très légère elle ne gênerait pas même avec les petites ressources de l'embarqué, je l'ai donc placé à la fin d'enregistrement.cpp pour qu'elle soit appelé à chaque fois qu'enregistrement.cgi l'est.


```

258         cout<<"\n\t\t\n";
259         system("sqlite3 /home/pi/bdd/Enregistrement.db < /home/pi/ShelScript.txt");
260
261     return 0;
262 }

```

Ensuite il me fallait un moyen de récupérer bdd.txt sur le mobile de l'opérateur. Au départ j'ai cherché une solution en javascript mais au fur de mes recherche je me suis rendu compte que HTML5 avait déjà les outils dont j'avais besoin, il y a notamment la balise `<a>` qui possède l'attribut "download". J'ai donc créer un fichier de test qui très bien fonctionné.

```

<html>
  <head>
    <title>TODO supply a title</title>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
  </head>
  <body>
    <link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/font-
    <a href="/home/pi/test.txt" download><button>Download BDD</button></a>
  </body>
</html>

```

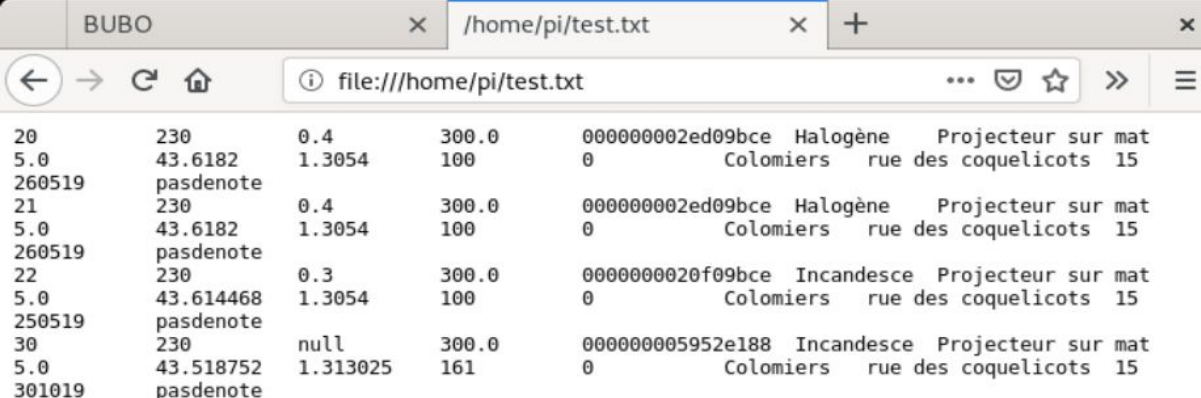
J'ai donc modifier le fichier index.html du projet pour qu'il corresponde à cette méthode. J'ai supprimer toute la partie JavaScript en rapport avec l'envoi de la base de données et j'ai modifier le bouton.

```

<div class="ui-block-b"><a href="/home/pi/bdd.txt" download><input type="button" id="envoibdd" value="Envoi bdd" ></a></div>

```

Seulement au lieu de télécharger le fichier, quand on appuyait sur le bouton il apparaissait dans un nouvel onglet.



The screenshot shows a web browser window with a single tab titled 'BUBO'. The address bar displays 'file:///home/pi/test.txt'. Below the address bar, a table of data is visible. The table has 7 columns and 10 rows of data. The data appears to be a list of projectors with various attributes like power, location, and status.

20	230	0.4	300.0	000000002ed09bce	Halogène	Projecteur sur mat
5.0	43.6182	1.3054	100	0	Colomiers	rue des coquelicots 15
260519	pasdenote					
21	230	0.4	300.0	000000002ed09bce	Halogène	Projecteur sur mat
5.0	43.6182	1.3054	100	0	Colomiers	rue des coquelicots 15
260519	pasdenote					
22	230	0.3	300.0	0000000020f09bce	Incandesce	Projecteur sur mat
5.0	43.614468	1.3054	100	0	Colomiers	rue des coquelicots 15
250519	pasdenote					
30	230	null	300.0	000000005952e188	Incandesce	Projecteur sur mat
5.0	43.518752	1.313025	161	0	Colomiers	rue des coquelicots 15
301019	pasdenote					

Pensant que le problème venait mon installation j'ai demandé à Mme Bouyssonnade de tester la solution sur la Raspberry, mais il s'est passé la même

Doumenc Thomas

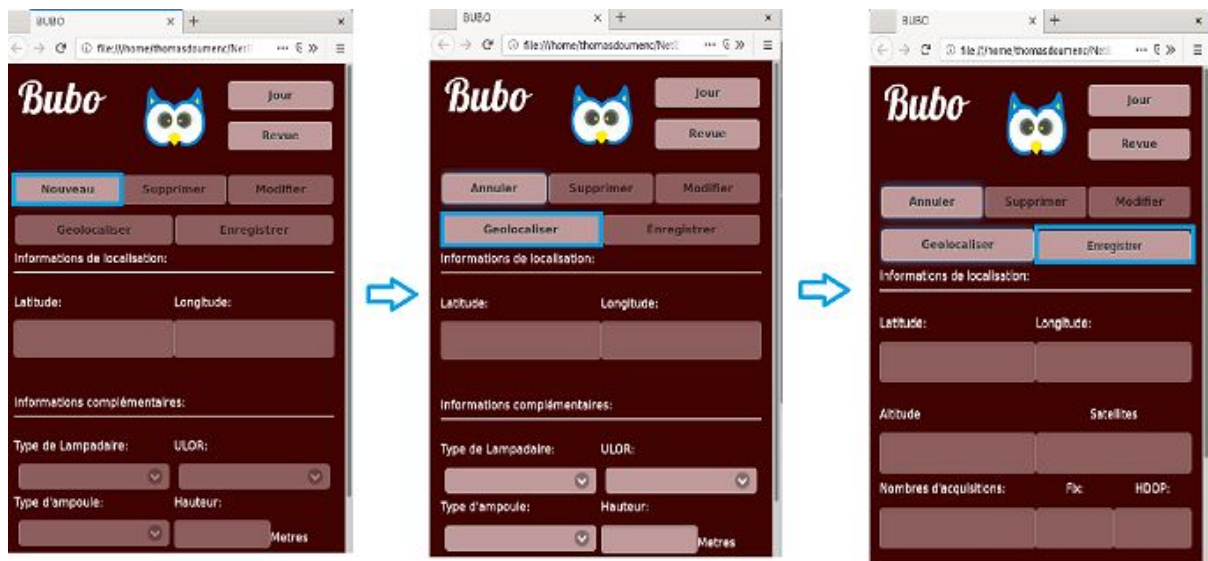
chose elle m'a alors aidé à trouver d'où venait le problème et m'a permis de me rendre compte que le problème venait de la bibliothèque

"`jquery.mobile.min.js`" qui pose problème. Celle ci pense que le chemin est un lien vers un page et l'ouvre donc dans un nouvel onglet. Pour régler le problème il faut simplement ajouter le code ci dessous pour désactiver l'ajax.

```
<a href="bdd.txt" data-ajax="false" download>
```

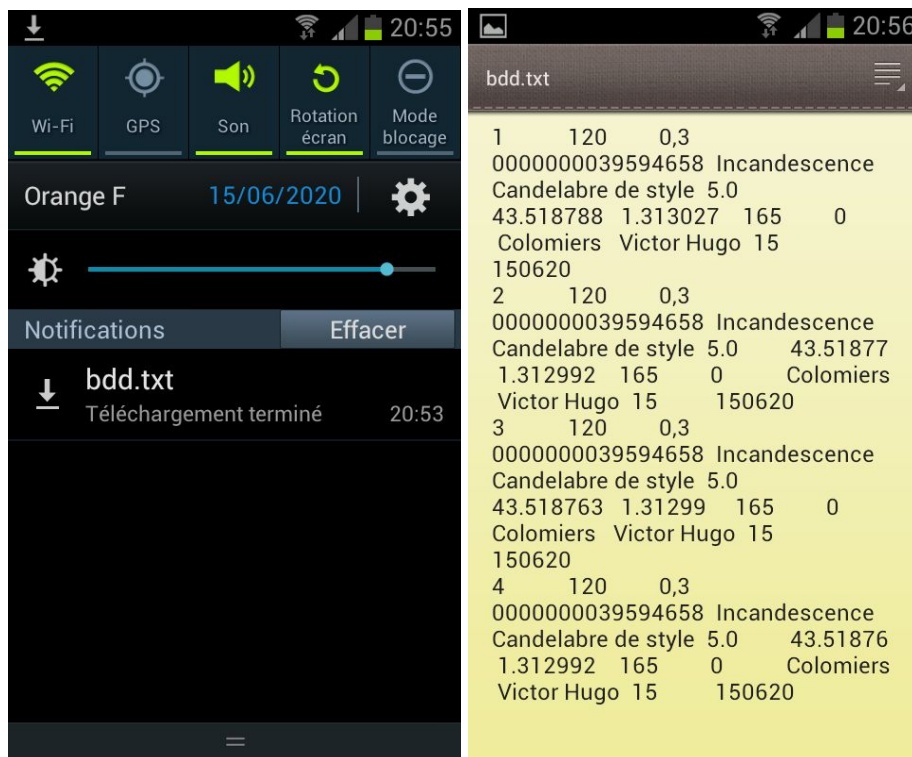
Test de mise en oeuvre

Tout comme pour le problème de l'altitude il faut commencer par agir sur un enregistrement



Ensuite on appuie sur le bouton de "Export bdd"

Le téléchargement se lance alors et on a bien le fichier.



Suivi d'activité

Semaine	Nom du développeur	Activité
Sprint n°1 Début : 03/02/2020 Fin : 06/03/2020		
Du 27/01/2020 Au 30/01/2020	DOUMENC	<ul style="list-style-type: none"> - Découverte du projet. - Création d'un backup du projet. - Création d'un Google Drive commun de sauvegarde et partage de données.
Du 3/02/2020 Au 6/02/2020	DOUMENC	<ul style="list-style-type: none"> - Validation des éléments fonctionnel. - Analyse de l'architecture . - Compréhension du besoin clients.
Scrum Meeting 05/02/2020		
Du 08/02/2020 Au 23/02/2020	DOUMENC	Vacances Scolaire.
	DOUMENC	
Du 24/02/2020 Au 27/02/2020	DOUMENC	<ul style="list-style-type: none"> - Compréhension de l'architecture.. - Compréhension du code pour la mise à l'heure.
Réunion avec le client 26/02/2020		
Du 02/03/2020 Au 05/02/2020	DOUMENC	<ul style="list-style-type: none"> - Mise à jour des exigences clients. - Réorganisation
Sprint n°2 Début 09/03/2020 Fin : 27/03/2020		
Du 09/03/2020 Au 12/03/2020	DOUMENC	<ul style="list-style-type: none"> - Recherche de solutions pour la mise à l'heure - Teste d'implémentation des solutions trouvées
Du 16/03/2020 Au 19/03/2020	DOUMENC	<ul style="list-style-type: none"> - Mise en place de la machine virtuelle à mon domicile - Création des diaporamas pour la revue de projet
Revue de projet n°1 18/03/2020		
Du 23/03/2020 Au 26/03/2020	DOUMENC	<ul style="list-style-type: none"> - Configuration des logiciels nécessaire comme NetBeans . - Récupération du projet
Scrum meeting 24/03/2020		
Sprint n°3 Début 30/03/2020 Fin 30/04/2020		
Du 30/03/2020 Au 02/04/2020	DOUMENC	<ul style="list-style-type: none"> - Configuration du code pour simuler le gps - Recherche de solutions

Du 04/04/2020 Au 19/04/2020	DOUMENC	Vacances Scolaire.
	DOUMENC	
Du 20/04/2020 Au 23/04/2020	DOUMENC	- Reconstruction de la machine virtuelle suite à un bug qui l'a détruite
Du 27/04/2020 Au 30/04/2020	DOUMENC	<ul style="list-style-type: none"> - Compréhension du code qui s'occupe de faire les enregistrement dans la base de données - Recherche de solutions - Validation de la tâche 1
Scrum meeting 29/04/2020		
Sprint n°4 Début 04/05/2020 Fin 22/05/2020		
Du 04/05/2020 Au 07/05/2020	DOUMENC	<ul style="list-style-type: none"> - Compréhension de l'architecture web du projet - Recherche de solution pour l'exportation de la base de données au format texte
Du 11/05/2020 Au 14/05/2020	DOUMENC	<ul style="list-style-type: none"> - Mise en place des solutions trouvées - Test du code
Du 18/05/2020 Au 21/05/2020	DOUMENC	<ul style="list-style-type: none"> - Test de validation. - Ecriture du dossier technique d'examen.
Sprint n°5 Début 25/05/2020 Fin 12/05/2020		
Du 25/05/2020 Au 28/05/2020	DOUMENC	<ul style="list-style-type: none"> - Ecriture du dossier technique d'examen. - Amélioration du code
Du 01/06/2020 Au 04/06/2020	DOUMENC	<ul style="list-style-type: none"> - Ecriture du dossier technique d'examen. - Documentation Du code
Scrum meeting 03/06/2020		
Du 08/06/2020 Au 11/06/2020	DOUMENC	<ul style="list-style-type: none"> - Ecriture du dossier technique d'examen. - Documentation Du code
Fin du projet 12/06/2020		

Bilan

Travailler sur BUBO a été une super expérience, j'ai pu découvrir ce que signifie réellement travailler sur un projet en groupe, ce qui me sera très utile à l'avenir.

J'ai pu mettre en oeuvre une bonne partie de ce que j'ai appris au cours des deux dernières années et même approfondir mes connaissances notamment en C++ et programmation objet. J'ai aussi pu mettre à l'épreuve mes compétences en terme d'organisation.

Pour finir je tiens à remercier Mme Bouyssonnade qui nous a accompagné tout au long de ce projet malgré les conditions spéciales de cette fin d'année.