Local Application Administrator Guide

Welcome to use this local application.

Please have a look at README carefully, which is this guide and help you to install and use the local application on raspberry pi as administrator.

Thanks for the guidance of Dr. Antoniadis to make this local application so well. If you would like to run it on raspberry pi, please contact him for the **raspberry pi configuration guide.**

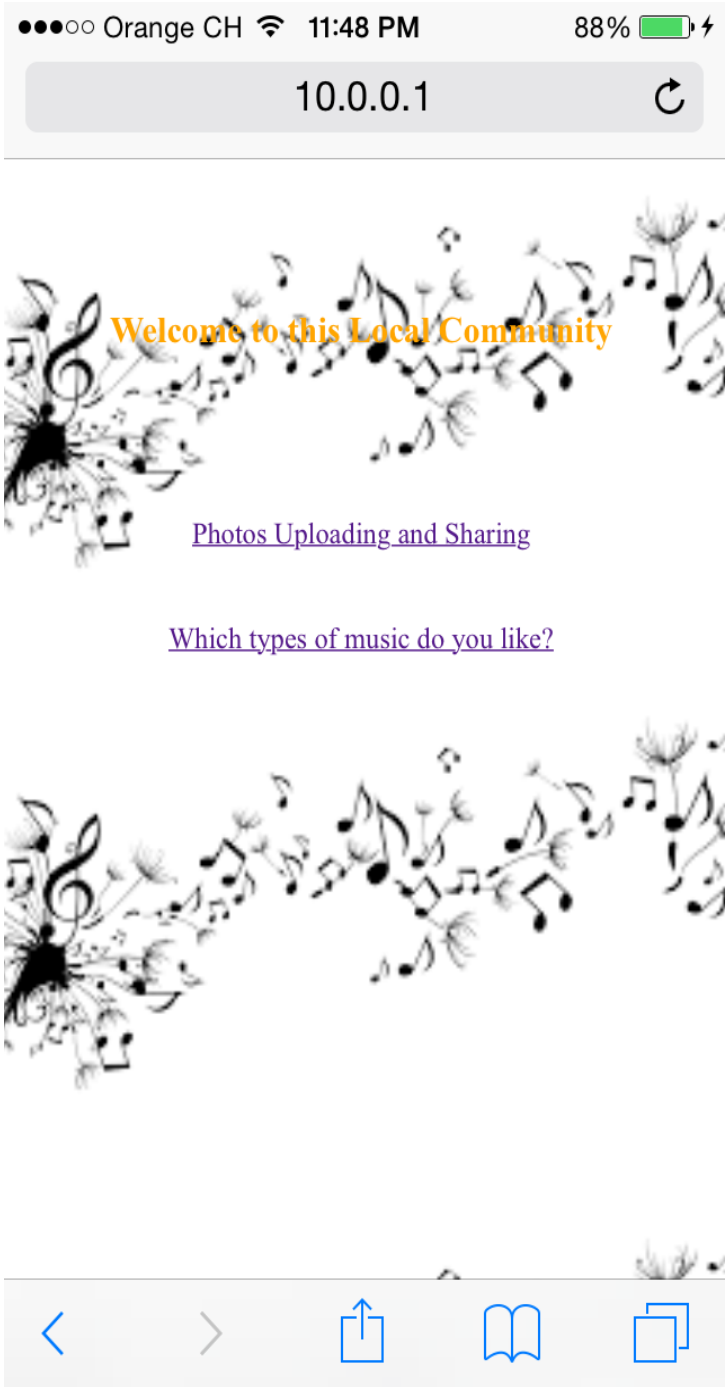
Thanks for the source code provided by the two websites below, which is modified to suit this local application.
http://www.foliopages.com
http://www.phpkobo.com/ajax_poll.php

Please put photo_upload, questionnaire, server, index.html and background.jpg in the /var/www/ directory of your raspberry pi and also in the localhost/ directory of your own computer as well.

Briefly, this local application consists of three folders for sharing the photos, the questionnaire and the server. In the **offline mode** described by the **raspberry pi configuration guide, all the traffic of the users will be redirected to the captive portal of the raspberry pi. Enjoy it and have fun!!!**

Note: We are not responsible for the photos have been uploaded, the questions chosen by the ones who has taken the role as administrator of this local application and the messages left on this local application.

Here is the interface of local application



The photo_upload folder contains the scripts with photos uploading and sharing for the users, who can upload the photos with extensions: ".jpg",".jpeg",".png",".gif",".JPG",".JPEG",".PNG",".GIF". In addition, the users can also leave messages as well at the same page for a better communication locally (This idea is inspired by stupid forum). The comments of the code can be found in detail in the

scripts. In order to enable its function, we need to give the permissions of the corresponding folders by the commands below.

```
sudo chmod 777 /var/www/photo_upload/album
sudo chmod 777 /var/www/photo_upload/album/*
sudo chmod 777 /var/www/photo_upload/forum
sudo chmod 777 /var/www/photo_upload/forum/*
```

You may also change the number of photos you want to display in the script folio-gallery.php, which is set 30 initially.

```
<?php

// error_reporting (E_ALL ^ E_NOTICE);
// photo gallery settings
$mainFolder    = 'album';   // folder where your albums are located - relative to root
$itemsPerPage  = '30';       // number of images per page, you can change according to your need.
```

Here is the interface of Photo Uploading and Sharing

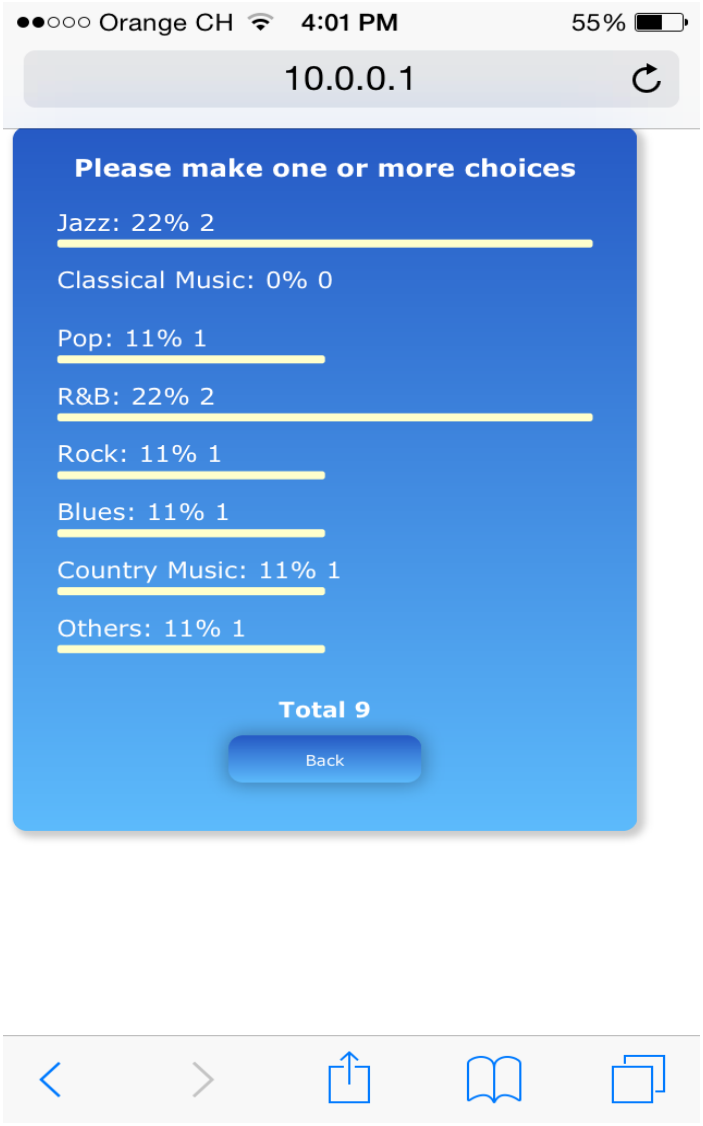
The questionniare folder contains the scripts for making a questionnaire and records the voting results. In order to enable its function, we need to give the permissions of the corresponding folders by the commands below.

```
sudo chmod 777 /var/www/questionnaire/app.data
sudo chmod 777 /var/www/questionnaire/app.data/*
```

The results file is stored below with the associated raspberry pi id, which will be explained later when it comes to the configuration of the raspberry pi. The server folder contains the code to upload this file to the ftp server after 10 seconds of each voting.

/questionnaire/app.data/poll-multi-choice.def/votes_Rpi1.txt

Here is the interface of questionnaire.



As administrator, you can change the title of the questions, the poll items and the name of the data file as the ways indicated below.

For **changing the title** of the quesionnaire you have to go to the script : ***/questionnaire/poll-multi-choice/class.inc.php***

At the beginning of the script, you will find the code to assign the title of the questionnaire.

```
//-- Poll Title
$poll->attr( "title", "Please make one or more choices " );
```

You can change the title by modifying the string inside the double quote at the right side of the title. For example, in this case, the title of the quesionnaire is “Please make one or more choices”.

For **changing or adding the poll items** of the questionnaire, you can go to the same script just as before. The code for assigning the questions is just below the Poll Title.

```
//-- Poll Options
$poll->addItem( "Jazz" );
$poll->addItem( "Classical Music" );
$poll->addItem( "Pop" );
$poll->addItem( "R&B" );
$poll->addItem( "Rock" );
$poll->addItem( "Blues" );
$poll->addItem( "Country Music" );
$poll->addItem( "Others");
```

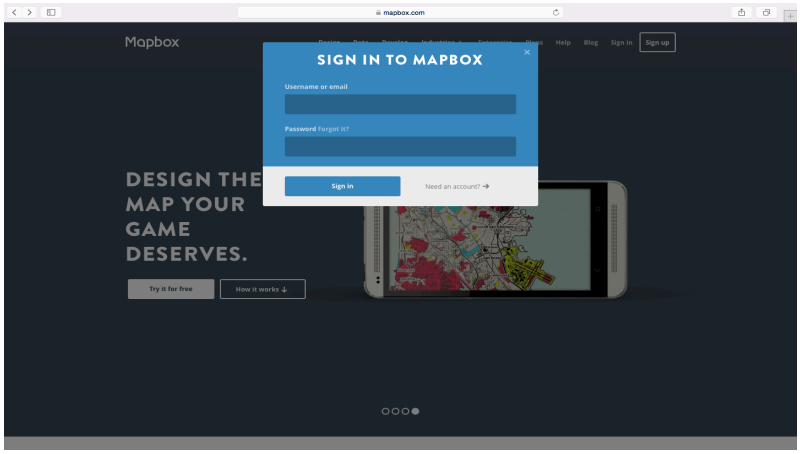
By following the syntax from the example, you can delete or add a poll item.

For changing the name of the data file associated to the corresponding raspberry pi id, you have to go to the script: ***/questionnaire /app.ajax-poll/include/CPoll.inc.php, you will find the code for assigning the name of this file at line 221 as below:***

```
function getDataFilePath() {
        return $this->prt->getDataFolderPath() . "votes_Rpi1.txt";
}
```

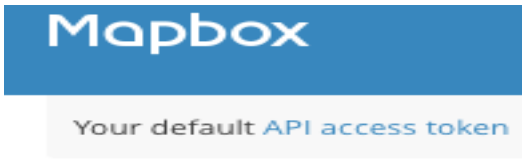
The data file **votes_Rpi1.txt** is the voting results associated to the raspberry pi with id **Rpi1.** Therefore, the voting results associated to the raspberry pi with **id Rpi2 is votes_Rpi2.txt so on and so force. Please follow this format of the data file strictly, we only change the number to indicate the corresponding raspberry pi.**

The server folder contains the scripts for the registration of the raspberry pi on the map and see the voting results on the map.In order to use this server, you have to go to the Mapbox to register first. And then sign in to MapBox with your username and password.



Then click Data like below.

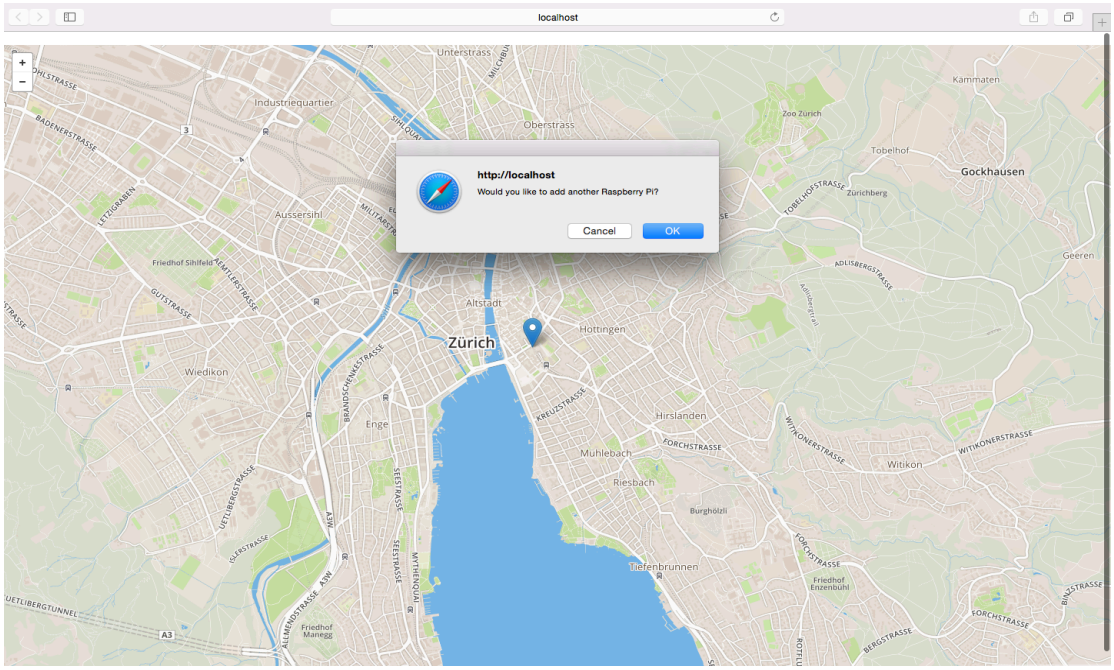
You will find you Default API Access Token, which is a long string at the right side of the API access token.



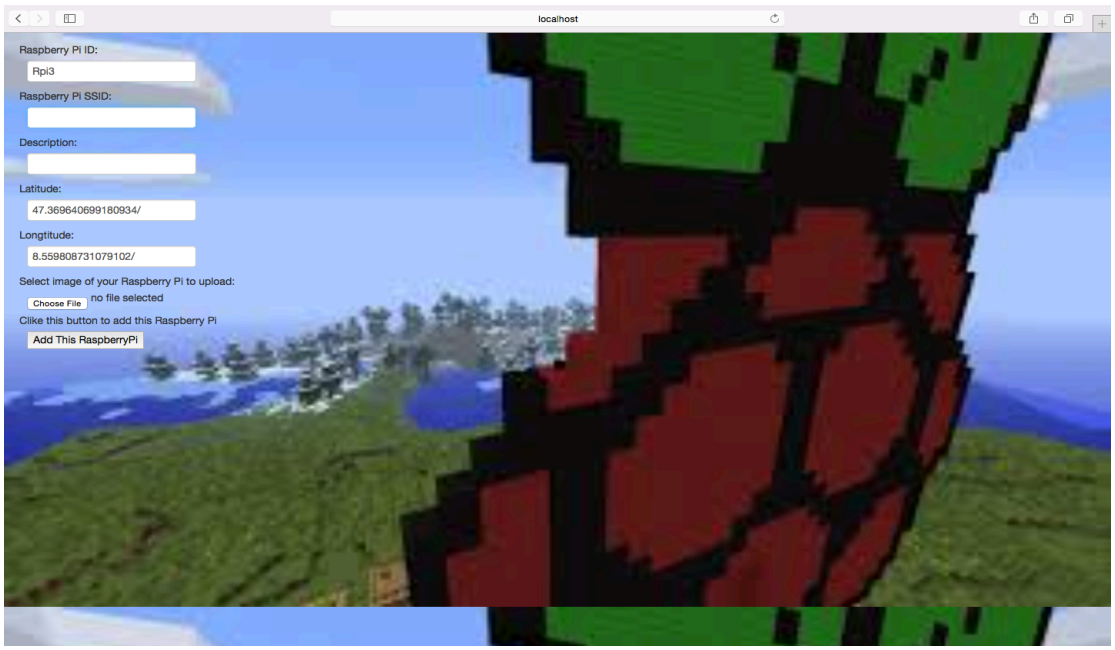
This token will be copied and paste into the script map_server.php at the location below, which is the yellow part.

```
<script>
L.mapbox.accessToken = '';

var mapboxTiles = L.tileLayer('https://{s}.tiles.mapbox.com/v3/panayotis.kbh9i6fk/{z}/{x}/{y}.png', {
            attribution: '<a href="http://www.mapbox.com/about/maps/" target="_blank">Terms &amp; Feedback</a>'
            });

var rpis = <?php echo json_encode($raspberries, JSON_PRETTY_PRINT) ?>;
var votes = <?php echo json_encode($results,JSON_PRETTY_PRINT) ?>;
```

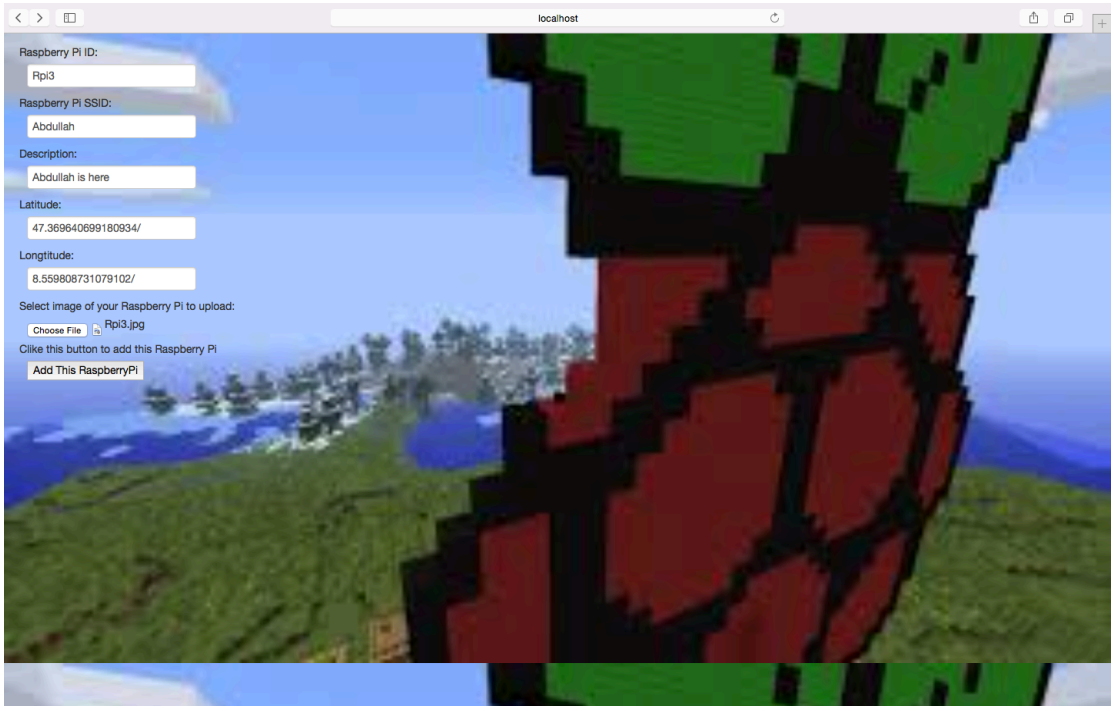
Now you can insert the raspberry pi from the localhost/server/map_server.php of your own computer as administrator. By clicking the map, you will find a window to ask you whether you will insert a raspberry pi or not as below. This page refreshes itself every 10 seconds to check whether there are something votes happens or not.



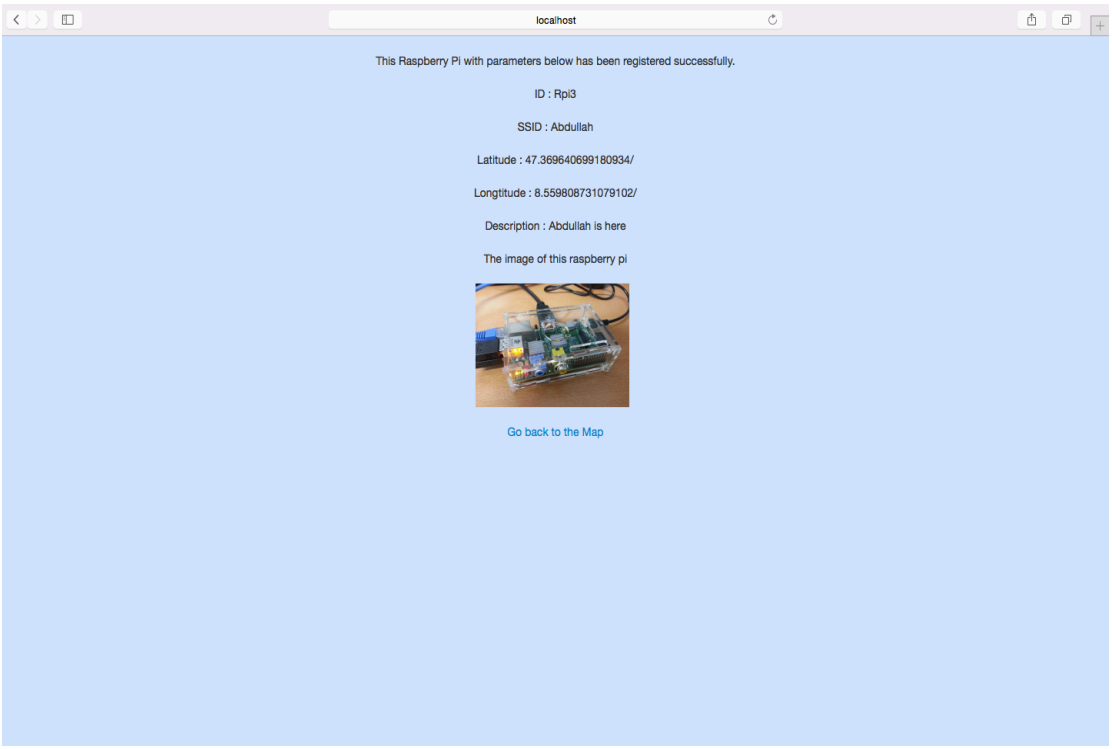
If you click **ok**, you will be directed to a page to insert the parameters of the raspberry pi as below.



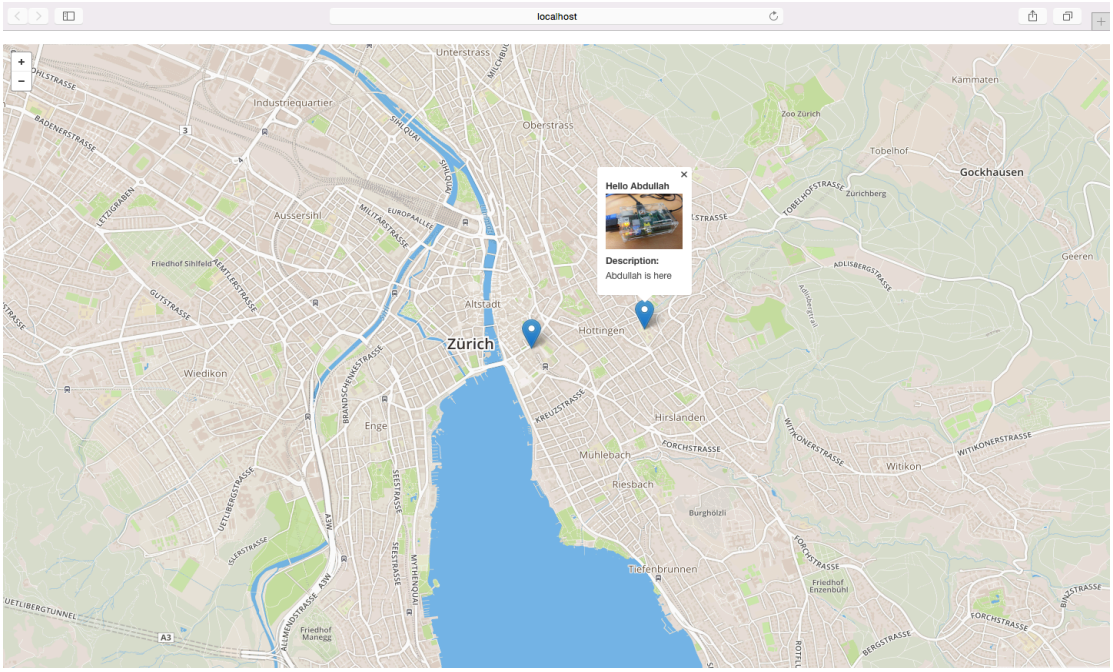
Pay attention, you just have to insert the **SSID, Description** and **upload the photo** of the raspberry pi. (***The photos of raspberry I should also have strict naming, which can be seen in the folder Rpi_photos***)The **id** of the raspberry pi is set as the index of the raspberry pi you are inserting. The **Latitude** and **Longitude** are passed from the position of the map you click to add the raspberry pi. **NICE!!!**



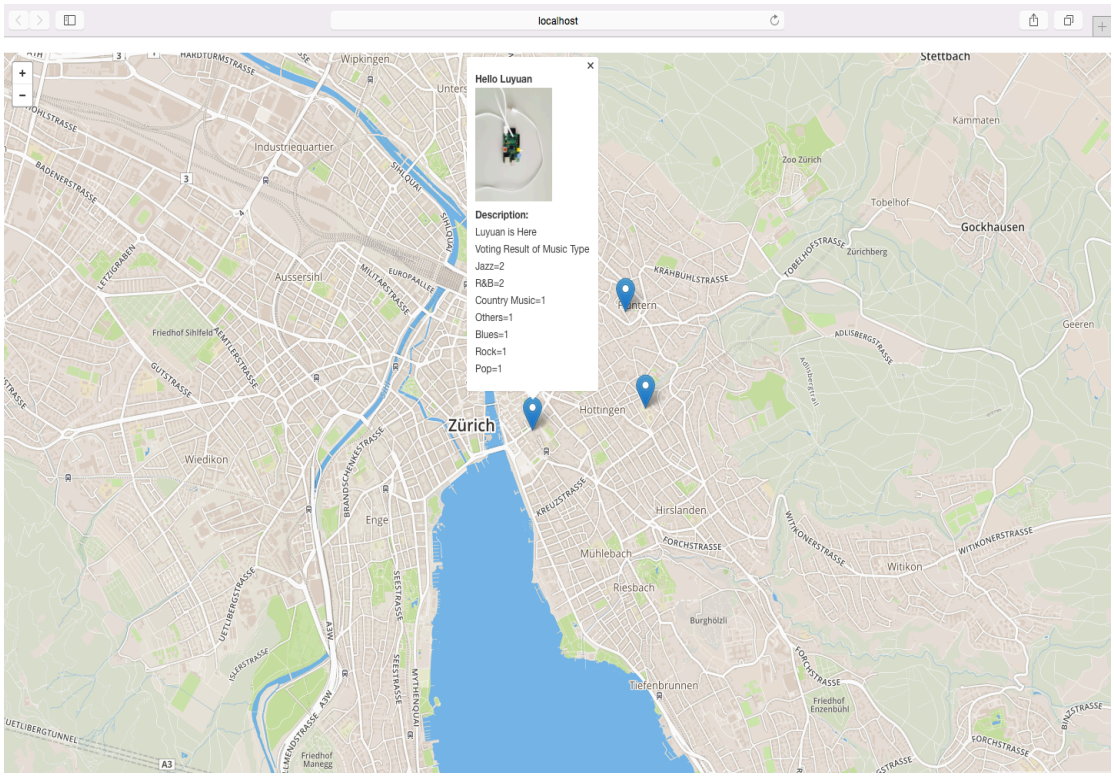
After successfully inserting the raspberry pi, you will be directed to a page where shows the information of the raspberry pi you have inserted.

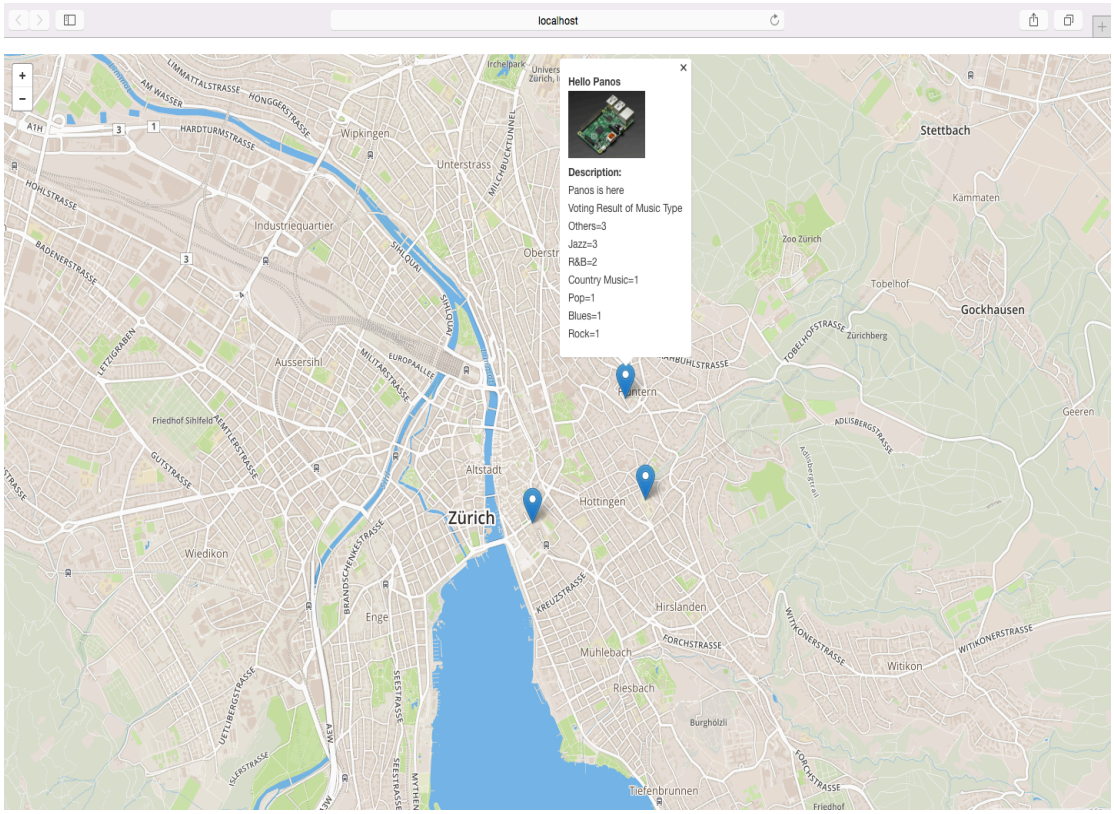


At the mean time, the photo of this raspberry pi will be uploaded to the ftp server(If the raspberry pi is connected to the internet). Then click go back to map, you will see the new raspberry pi on the map.



Then by following the **raspberry pi configuration**, you can find the way to assign the correct **SSID** to the raspberry pi. In our example, you can see nothing happens on the inserted raspberry pi, but for the previous two, you can see something happens there.





If you are offline, in order to processing the results of the data file, you have to download the data file from the ftp server, ***/www/server/votes/,*** then put it into the ***localhost/server/votes, clicking the raspberry pi on the map, you will see the results at the corresponding point.***

For seeing the results on the map, we have to add or delete the items in the script /www/server/map_server.php at line 203 according to the items of questionnaire we made by following the examples below.

```
+"<br>"+votes[key].n1
+"<br>"+votes[key].n2
+"<br>"+votes[key].n3
+"<br>"+votes[key].n4
+"<br>"+votes[key].n5
+"<br>"+votes[key].n6
+"<br>"+votes[key].n7
);

//If you have more items
//Add in this way
//+"<br>"+votes[key].n? ?goes the number you want
```

The offline installation guide for this local application on the raspberry pi is finished here. The online installation will be done soon !!!