**Installation Guide**

**CITADEL…on the move**

**Mobile Application Templates: ‘Points of Interest (POIs) in the city’**

**Project co-funded by the European Commission within the ICT Policy Support Programme**

Table of Contents

[CITADEL…on the move 1](#_Toc342055384)

[1. CITADEL Mobile Application template 3](#_Toc342055385)

[1.1 Structure of a template 3](#_Toc342055386)

[1.2 Deploying a template in 5 steps 4](#_Toc342055387)

[1.3 Set up the backend 5](#_Toc342055388)

[1.4 Changing the dataset 5](#_Toc342055389)

[1.4.1 Change using the backend 5](#_Toc342055390)

[1.4.2 Change without backend 6](#_Toc342055391)

[1.5 Settings of the application 6](#_Toc342055392)

[1.6 Prerequisites 7](#_Toc342055393)

# CITADEL Mobile Application template

## Structure of a template

A template in the context of the CITADEL project is actually a package containing all the necessary files that will allow a citizen developer to easily deploy it and afterwards extend it. The technology stack of a template consists of the following widely used technologies:

* HTML5
* Javascript
* PHP
* MySql
* JSON

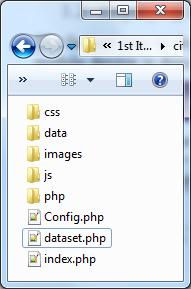
It should be also noted that the jQuery Mobile javascript framework is also used in the templates. There has been an effort to limit the use of the framework for visualization purposes and for the navigation between the pages only. Data retrieval and manipulation have been implemented using simple javascript functions.

Inside the template all files are organised under the following folder structure:

1. php – contains all php files
2. js – contains all javascript files
3. css – contains all css files
4. images – contains the images used
5. data – contains datasets in JSON format (‘.json’ files) as well as the database dump
6. doc – contains the installation and programming guides

The file serving as the home page of the application is “index.php” and is located outside of all the above folders and inside the root folder of the template. File “Config.php” includes the settings for the application as presented in Section 1.5. The “dataset.php” file contains the code responsible to load the dataset from the .json file or the database.

The folder contents are depicted in Figure 1.



Figure

## Deploying a template in 5 steps

The fastest way to deploy a template and view it in the web browser of a desktop or a mobile device is described in the following steps:

|  |
| --- |
| Deployment in 5 STEPS |
| 1. Download the template **citadel-pois-template.zip** |
| 1. Unzip the contents in a file named **citadel-pois-template.** This file should contain all the folder structure described in the previous paragraph |
| 1. Put the **citadel-pois-template** folder inside the web directory of your local webserver |
| 1. Open a browser and point it at:   [http://localhost/](http://localhost/citadel-pois-template/index.php)**[citadel-pois-template](http://localhost/citadel-pois-template/index.php)**[/index.php](http://localhost/citadel-pois-template/index.php) |
| 1. The template application is up and running with the default sample dataset |

Things to notice:

* You should not change the folder names or the folder structure described in the previous steps in order to successfully deploy the application.
* If you want to also access the application from a mobile device connected to the same network as the desktop machine, then you should modify the url given in step 4 as follows: http://<your-current-IP[[1]](#footnote-1)>/**citadel-pois-template**/index.php.
* There are a number of browser plugins allowing you to see how the application looks when accessed by different-sized devices (e.g. Mobile Resizer for Chrome). Note that the looks might eventually be slightly different in real mobile devices.

## Set up the backend

The process described in the previous section will result in an application set up with the default included datasets. These sample datasets are plain text files containing data in the JSON format. They can be found under the ‘data’ folder with the extension ‘.json’. In order to select a different dataset (e.g. from another city) or to change the default functionalities of the application you should setup the backend of the application. A prepopulated mysql database is provided in the form of database dump containing all the necessary info about the database schema and initial data. This information is in the ‘citadel-backend.sql’ script file which can be found under the ‘mysql’ folder.

The following steps must be followed to connect the database with the application:

|  |
| --- |
| Setup of the backend |
| 1. Run the ‘citadel-backend.sql’ script in a MySql server instance[[2]](#footnote-2). This will create a new database named “**citadel-backend**”, prepopulated with sample data. |
| 1. Open “Config.php” and set: “useDatabase = true” |
| 1. Reload the application. It should now show data coming from the database and new functionalities will be now available. |

## Changing the dataset

### Change using the backend

Once the backend of the template is created, new datasets from other cities can be easily imported. These datasets should follow the appropriate CITADEL common representation format. For the ‘Parking Lots’ template, the dataset must follow the CITADEL common POI format. More information about the format and how to convert datasets can be found in the Programming Guide found in the ‘doc’ folder.

When a new dataset is imported, the map centre and the dataset id must be updated in order to display the data of the new city. These can be configured by setting the appropriate values of ‘**MAP\_CENTER\_LATITUDE**’, ‘**MAP\_CENTER\_LONGITUDE**’ and ‘**DATASET\_ID**’ inside the ‘Config.php’ file.

### Change without backend

It should be also noted that the ‘data’ folder also includes .json files containing data from different cities. So, before setting up the backend you could switch between these .json files and watch the application running in different cities. Changing the ‘**DATASET\_FILE**’, ‘**MAP\_CENTER\_LATITUDE**’ and ‘**MAP\_CENTER\_LONGITUDE**’ settings is required in order to switch to a different city without using the backend facility.

## Settings of the application

The list of settings that can be found in the “Config.php” file and their meaning are presented in the table below:

Table 1: Mobile Application Settings

|  |  |  |
| --- | --- | --- |
| setting | Default value | meaning |
| USE\_DATABASE | false | Indicates weather the template will use data from the database or not |
| DATASET\_FILE | POI\_gent.json | The name of the .json file that will be loaded if no database is used. |
| DATASET\_ID | 1 | The dataset id is created when a dataset is imported in the backend. The template comes with a prepopulated database which includes the same dataset as the default .json file. |
| DATASET\_URL | dataset.php | The url to the service that retrieves the dataset |
| HTDOCS\_ROOT |  | The root directory of the web server. There is no standard path for this so it should be changed to match the current root directory. |
| BASE\_DIR | Citadel-Pois-Template/ | This is the root folder of the application. This should not be changed in no backend is used. |
| CLASSES\_DIR | php | This is the folder containing the php files. This should be not changed if the default folder structure is followed. |
| MAP\_CENTER\_LATITUDE | 51.033261 | The latitude of the centre of the map. This should be the centre of the city in which the dataset refers to. |
| MAP\_CENTER\_LONGITUDE | 3.726488 | The longitude of the centre of the map. This should be the centre of the city in which the dataset refers to. |
| MAP\_ZOOM | 16 | The initial zoom level of the google map. |
| DB\_NAME | citadel | The name of the database. It should not be changed if the database dump is used to create the database |
| DB\_USERNAME |  | The username of a user with full rights to the database |
| DB\_PASSWORD |  | The password of the user |
| DB\_HOSTNAME | 127.0.0.1 | The default host name |
| DB\_PORT | 3306 | The default port |

## Prerequisites

Running the mobile application templates requires a minimum of software elements installed to your machine. A web server and PHP support are needed in order to be able to deploy the template using the 5 steps described in Section 1.2. MySQL database server is required for the backend setup.

1. **Web server**  
   In order to serve the generated pages to your browser, the template requires some web server software. The most common choice would be the [Apache](http://httpd.apache.org/) (<http://httpd.apache.org/>) web server, available at [the official download page](http://httpd.apache.org/download.cgi) (<http://httpd.apache.org/download.cgi>).
2. **PHP support**  
   [PHP](http://www.php.net/) (<http://www.php.net/>) is the programming language in which the template is written and is required to run the template application. It can be downloaded at [the official download page](http://www.php.net/downloads.php) (<http://www.php.net/downloads.php>). You should choose the download that is appropriate for the operating system of your machine, e.g. for windows the easiest way is to download the installer.
3. **MySQL**The backend of the template requires the use of a database and [MySQL](http://www.mysql.com/)(<http://www.mysql.com/>) is the chosen database server. It can be downloaded at [the official download page](http://www.mysql.com/downloads) (<http://www.mysql.com/downloads>).

The above technology stack is also called a WAMP (**W**indows **A**pache **M**ySQL **P**HP) environment (LAMP in linux). There are a number of software packages that perform the installation and configuration of this environment in one place, without having to download and set up the different elements separately. Two of the most popular are [XAMP](http://www.apachefriends.org/en/xampp.html) (<http://www.apachefriends.org/en/xampp.html>) and [WAMPServer](http://www.wampserver.com/en/) (<http://www.wampserver.com/en/>).

1. In Windows you can find the current IP by opening a command window and typing ‘ipconfig’ [↑](#footnote-ref-1)
2. If you are using the command line of my sql you can import the database dump by executing the following command: “mysql -h hostname -u user --password=yourpassword databasename < citadel-backend.sql” where ‘user’ is the name of the mysql user and ‘yourpassword’ is the password of the user. You could alternatively use some graphical mysql admin tool like [phpmyadmin](http://www.phpmyadmin.net/home_page/index.php) and use the ‘import’ functionality to load the database dump. [↑](#footnote-ref-2)