Deliverable D2 - Know Your City App Technical Documentation.

Expert contract number C391977 – Development of the Know Your City App to be demonstrated at the GEOSS Plenary meeting

Luis Rodríguez Pupo, Carlos Granell

The deliverable D2 provides technical documentation about the Know Your City app, according to the requirements included in the expert contract number C391977. This deliverable includes 1/ an installation manual of the Know Your City app, and 2/ an operations (user) manual. Where appropriate, the document provides links to external, public resources such as videos and presentations about the app to complement and enrich the content of the present deliverable.

1. Installation manual

The installation manual specifies the required software libraries needed to operate the application (i.e. dependences) and the steps to install and deploy a live instance of the Know Your City app.

The Know Your City web app was implemented by combining and integrating a set of open source technologies. At the client side, the app uses HTML and Javascript in conjunction with the Polymer framework version 1.0 for creating web components. In short, Polymer allows developers to modularize a large web application into a list of simple, reusable web components. At the server side, the app is based on restful web services, using the corresponding Java libraries, which permit easy access to and retrieve of data stored in backend databases. In particular, such database is an instance of PostgreSQL plus PostGIS.

In summary, the list of technologies used in the Know Your City app are:

- HTML (for self-develop code).
- Javascript (for self-develop code) and the JavaScript-based D3 library (http://d3js.org/).
- Polymer 1.0 (https://www.polymer-project.org/1.0/).
- Java REST API library that is called Jersey (https://jersey.java.net/). Resulting RESTful services require a web server container like Apache Tomcat (http://tomcat.apache.org/).
- PostgreSQL version 9.4 (http://www.postgresql.org/).
- PostGIS version 2.1 (http://postgis.net/).

Next we detail the contained components, libraries and services that form the Know Your City app, and how these are related to each each. Figure below shows an architectural view of the app divided in three traditional layers:

- a client web application, packaged as a single file called mygeosswebapp.war,
- middleware layer where RESTful web services reside, packaged as a single file called mygeossservices.war file, and
- a database called mygeossdb (Postgres database).

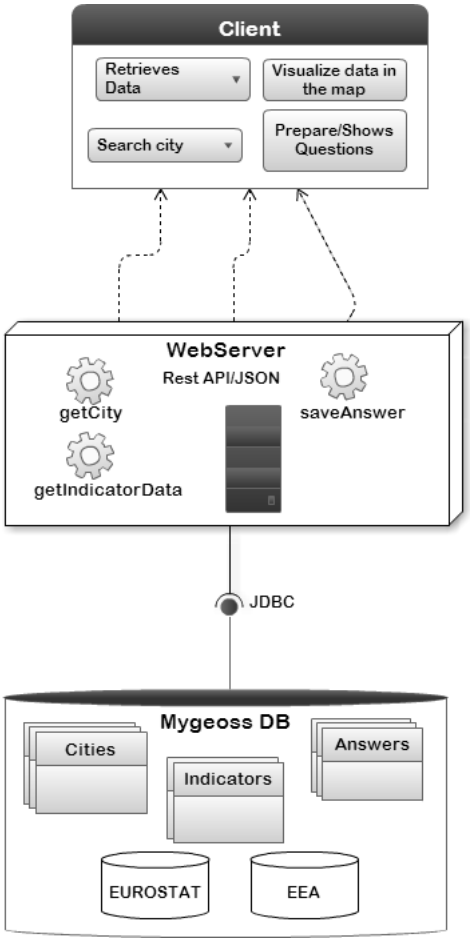

Figure: 3-layer architecture and contained components of the Know Your City app.

One way to deploy the app is to specify each step to install all dependencies required to deploy the war files above. For instance, a running instance of PostGres and Apache Tomcat and so on would be needed prior to deploy the app itself. Fortunately, this is no longer the case for the Know Your City app. To ease as much as possible the distribution and deployment of the app, we have used a recent technology called Docker (https://www.docker.com/). Docker greatly eases the deployment and setup of an application and all its dependencies by executing a few commands. Nevertheless, to do so, it is still necessary to previously install Docker in the host machine where the application will be deployed. For instructions on how to do this, please check out https://docs.docker.com/installation/.

The concept of images is central to Docker. Docker images are similar to the concept of reusable components. One may create a complex application by selecting appropriate components. The same is true for Docker images. One can set up a given application by specifying the corresponding Docker images in a configuration file. In addition, the docker engine smoothly manages the installation of the dependencies of the application. For the deployment of the Know Your City app, we have set up a couple of images in a public repository in the DockerHub web site:

- Lrodriguez2002cu/mygeossservices: This image contains the RESTful web services (mygeossservices.war) and the client web application (mygeosswebapp.war).
- Lrodriguez2002cu/mygeossdb: This image contains the database (mygeossdb) and include a volume for accessing the data.

In practical terms, apart from having Docker installed on the host machine, we only need to run the following two commands to install and deploy the app (Suggestion: A .bat file can be created to run the below code in one go).

```
docker run --name mygeossdb -p 5432:5432 -e
POSTGRES_PASSWORD=postgres -d lrodriguez2002cu/mygeossdb
```

```
docker run --name mygeossservices -d -p 80:8080 --link
mygeossdb:db lrodriguez2002cu/mygeossservices
```

The first command sets up a preconfigured Docker image to deploying a PostgreSQL database with PostGIS, and loading data sets used in the Know Your City app into the fresh instance of the database. The second command runs a preconfigured Docker image to set up an instance of Apache Tomcat upon which the RESTful web services and the client web application will be deployed.

In order for Restful services can access and communicate with the database, it is necessary to specify valid credentials. In the docker commands above, the default password was 'postgres' but this can be changed/customized by using the following commands (changes are marked in bold):

```
docker run --name mygeossdb -p 5432:5432 -e
POSTGRES_PASSWORD=mysecretpassword -d lrodriguez2002cu/mygeossdb
```

```
docker run --name mygeossservices -d -p 80:8080 --link
mygeossdb:db -e DB_PASS=mysecretpassword
lrodriguez2002cu/mygeossservices
```

The Docker image lrodriguez2002cu/mygeossservices admits additional configuration parameters. Please refer to the file /docker/services/Dockerfile of the source code for more information (see table below).

Once the installation commands are successfully executed, the application should be deployed and running. It can then be accessed by typing the following URL at the address bar of a browser.

http://[host_name]/knowyourcityweb/index.html

The new instance of the Know Your City app should appear in a browser as it does in the figure below:

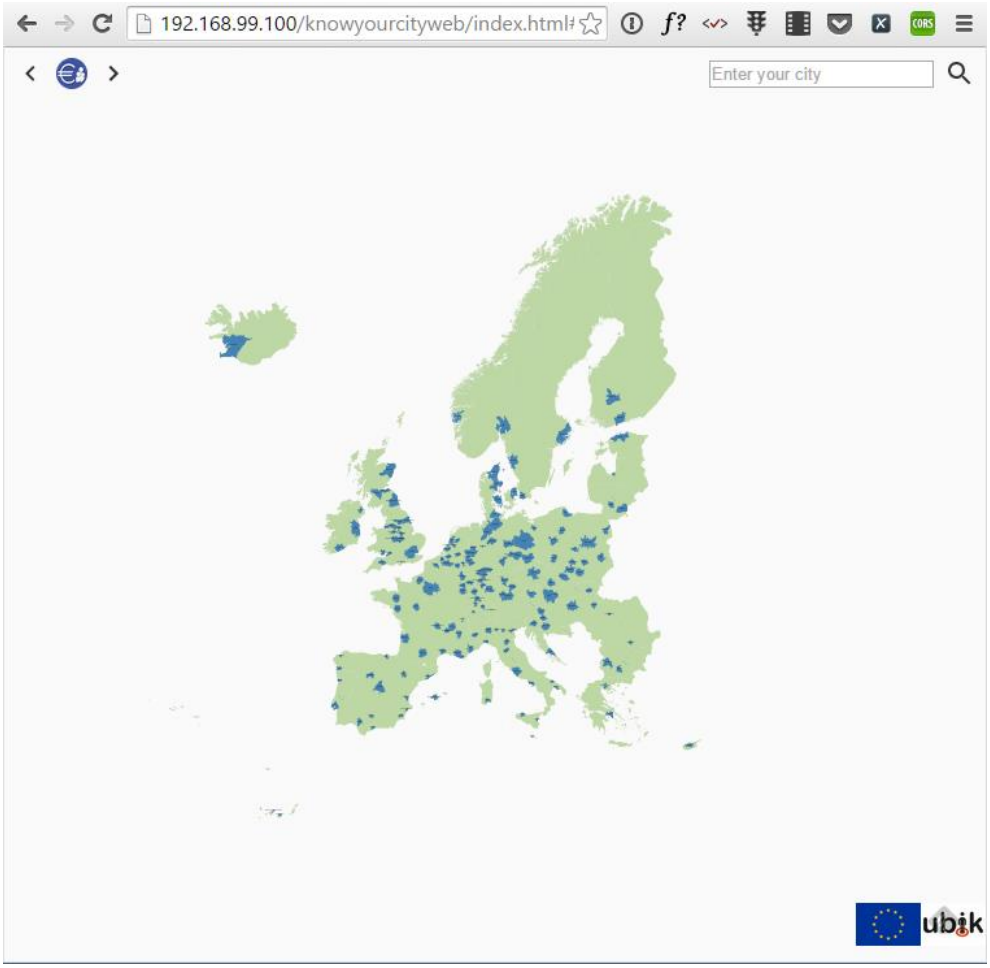

Figure: The Know Your City app home page.

Once the app is deployed in a local host machine, the root folder of the source code contains several subfolders that are being described in the table below. The source code of the contained libraries/artifacts/resources used by and/or developed for the Know Your City app can be found in the corresponding subfolder.

Subfolder name	Description
database\	It contains the all the components related to the MyGeoss database.
datatools\	It contains a small tool for the extraction of indicators from the Eurostat databases.
docs\	It holds documentation related to the project (i.e. Excel files, this document, presentation, original proposal docs).
mygeossservices\	It contains the source code of the backend services supporting the Mygeoss application. This project is managed using maven.
polymer-knowyourcity\	It contains the source code of front-end of the Know Your City application, i.e., a bunch of HTML and JavaScript files.

webapp\	A simple web application used for packaging the polymer-knowyourcity project into a war file for hosting the app on Apache Tomcat.
docker\	It contains all the artifacts for creating the docker images.

We strongly recommend to use Docker for easy installation and deployment of the Know Your City app. Nevertheless, as it is stated in the contract, we also provide the source code of the Know Your City app under EUPL licence at the following dropbox link: https://www.dropbox.com/sh/hzaou5mvvuvmgql/AABNqfS7M9iLaDM6PGtct0h1a?dl=0

2. Operations manual

The operations manual or user manual specifies the instructions to operate the Know Your City app and to use its functionalities.

Instead of describing the main features of the app as a sequence of figures (snapshots) and descriptive text, we created a couple of resources which may significantly reduce the learning curve to use and get familiar with the app.

The first resource is a running instance of the Know Your City app for testing and demonstration purpose. We have deployed it on the Amazon Cloud (AWS) and it is publicly available at the following URL:

http://ec2-52-88-98-222.us-west-2.compute.amazonaws.com/knowyourcityweb/index.html

Also, we have set up a customized domain name that links to the instance below. By typing http://knowyourcity.eu/ in a browser, a user also gets to the AWS instance of the app.

The second resource is a 5-minute video-manual that illustrates and explains the main features of the app. We demonstrated an interactive session with the public instance of the app at http://knowyourcity.eu/. This video-manual is reachable at:

https://www.youtube.com/watch?v=ypSWNOqD16c

Finally, it is worth to mention that we have developed specific web services to retrieve the anonymized user responses to quizzes. By calling the service below, all responses from users are returned in JSON format. This is especially interesting for later analysis in order to explore the level of users' awareness about the data sources used in the Know Your City app.

http://[host name]/mygeossservices/webresources/v1/mygeoss/answers

For example, in the public instance of the Know Your City app on AWS, user responses to quizzes are returned by querying the following service:

http://ec2-52-88-98-222.us-west-2.compute.amazonaws.com/mygeossservices/webresources/v1/mygeoss/answers