Carlos Ruano
Ernesto de Tovar
Mario Bizcocho

Alejandro Monteseirín
Camila Reyes
Ismael Lucas

Item 6: Cloud deployment

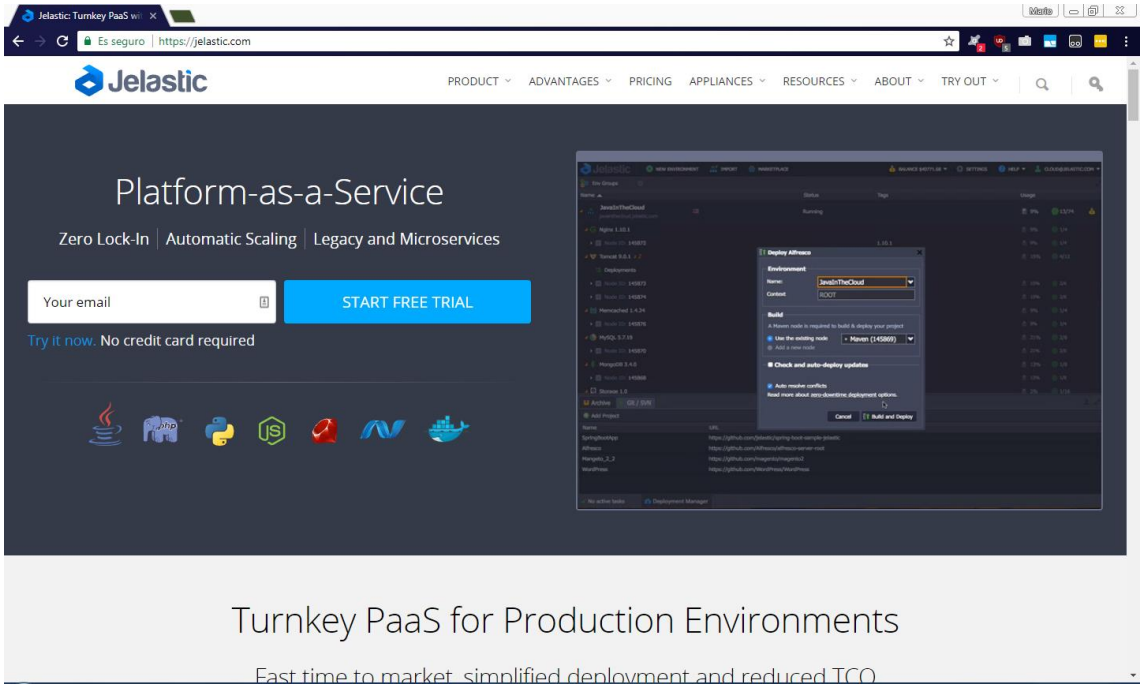
For this deliverable, we have chosen Jelastic as our deployment platform for the A+ requisites. Jelastic is an US-based cloud computing services provider, offering support for Java, Ruby, and Python, along with technologies PHP, Node.js, the .NET Framework, OpenShift Cartridges and Docker Containers. It offers only paid services, although a 14-day free trial is available, which is what we are using.

Its main rivals are Google App Engine, Amazon Elastic Beanstalk, Heroku, and Cloud Foundry. However, Jelastic doesn't impose limitations or code change requirements, and it offers automated vertical scaling, application lifecycle management, and availability from multiple hosting providers around the world. This allows for a simple deployment procedure consisting only on uploading the WAR file, setting up MySQL and establishing connections, which contrasts with other cloud service providers that offer an extremely complex deployment process and/or no free trial for some vital components (in Google's case, for instance, persistence servers are always paid).

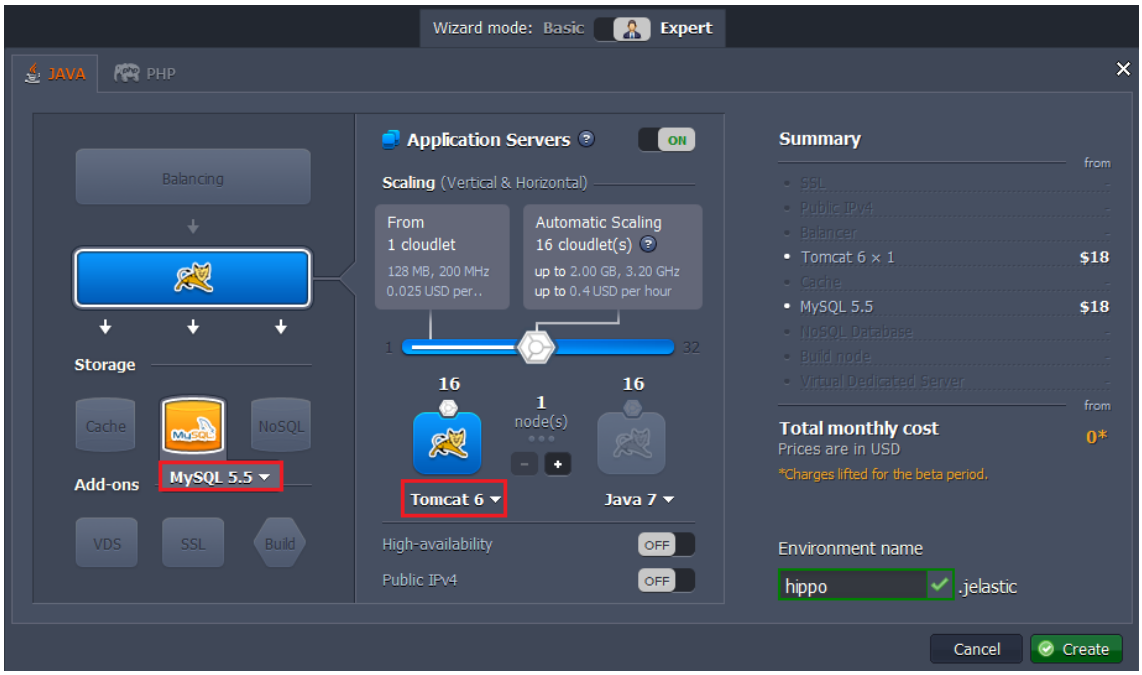
It must be said that, before setting for Jelastic, we tried deploying with OpenShift (which was extremely complicated, so we didn't finish), and App Engine, which required payment. Once we decided on a Jelastic deployment, though, there were no problems with the deployment itself and everything went absolutely flawless, despite a couple of bugs found in our code that don't have to do with the deployment operation.

Step 1: Register an account

Indeed, the first step to take is registering an account. To do this, all that must be done is accessing http://jelastic.com and entering your email, asking to start the free trial and then following the activation link. The free account should be operative in a short number of minutes.

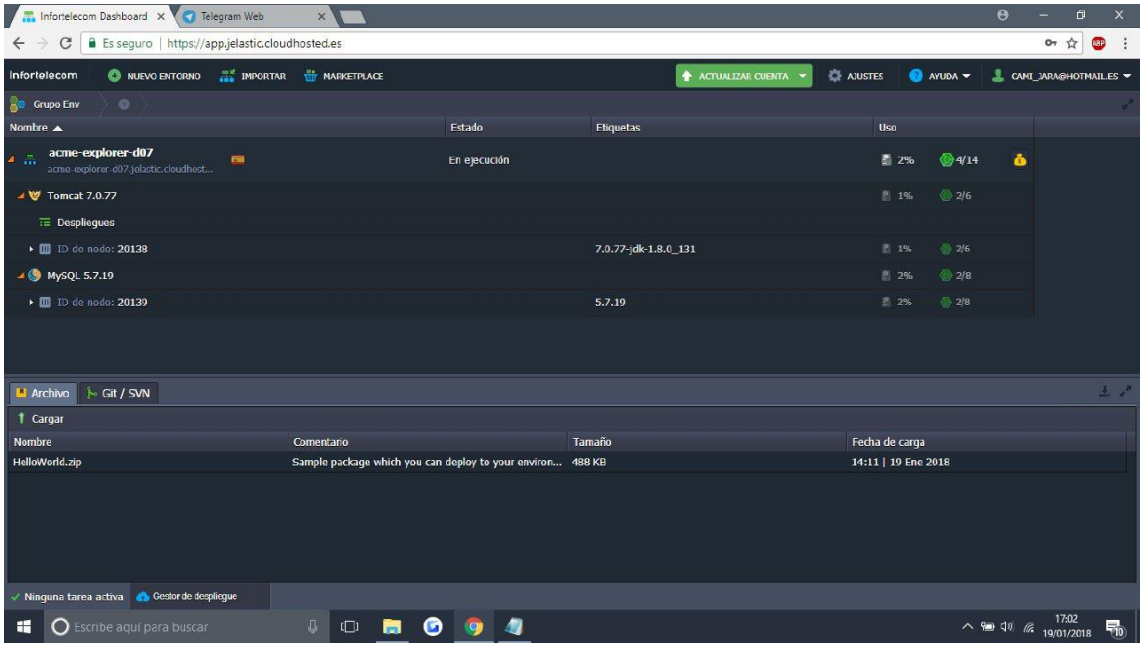


After doing this, we are greeted by a screen from where we can choose an application server to use. These range among all the technologies described above, and then some. For our Acme-Explorer, we need to deploy a Tomcat 7.0.77 and MySQL 5.7 combination, which, as it can be seen, is easy enough (please note that selected versions in the picture don't match our setup). This should be enough to create the basis for our servers, and now it is time to configure their content.

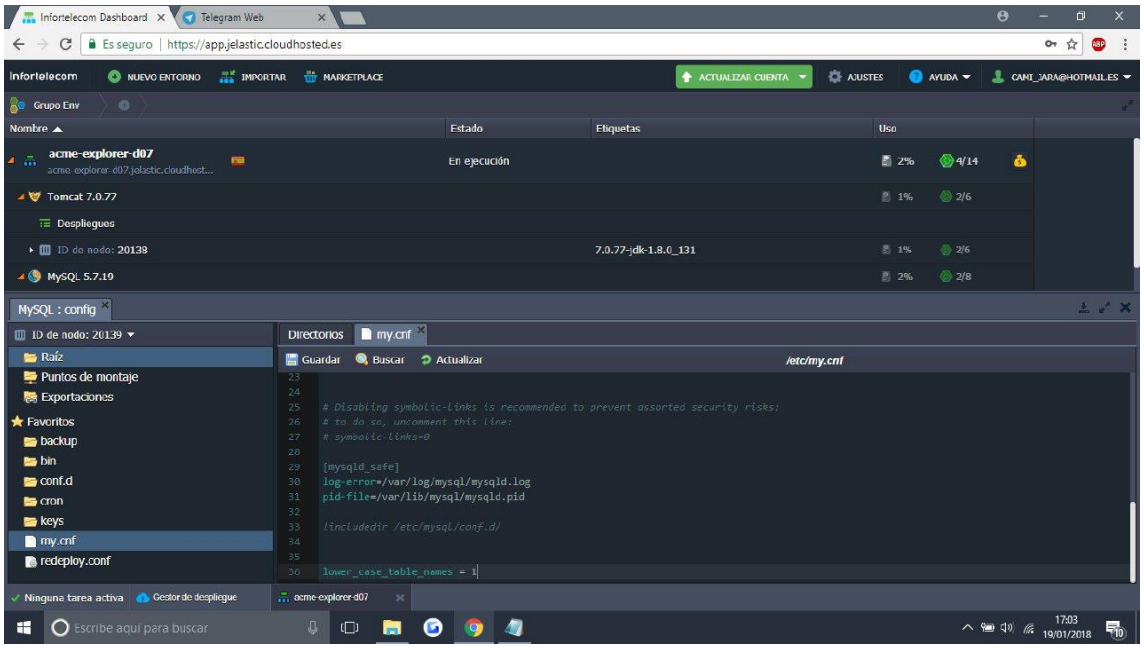


Step 2: Configure the servers and deploy

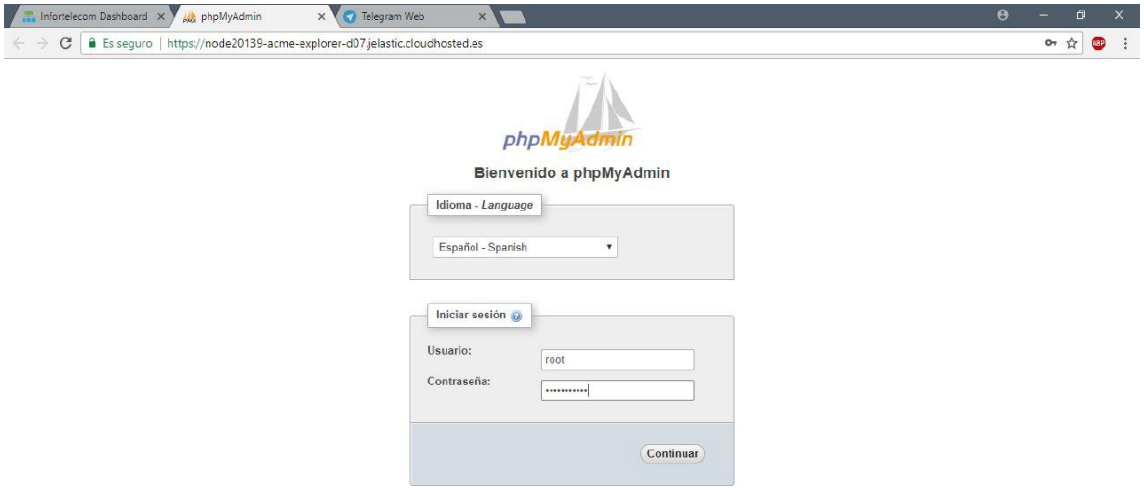
Once the servers are created, we are greeted by the Jelastic dashboard. This environment offers all the necessary tools to modify our project on the run without needing to redeploy from source in many cases. Some of the files created when our servers were set up must be modified to tell MySQL and Tomcat what and where our project is.

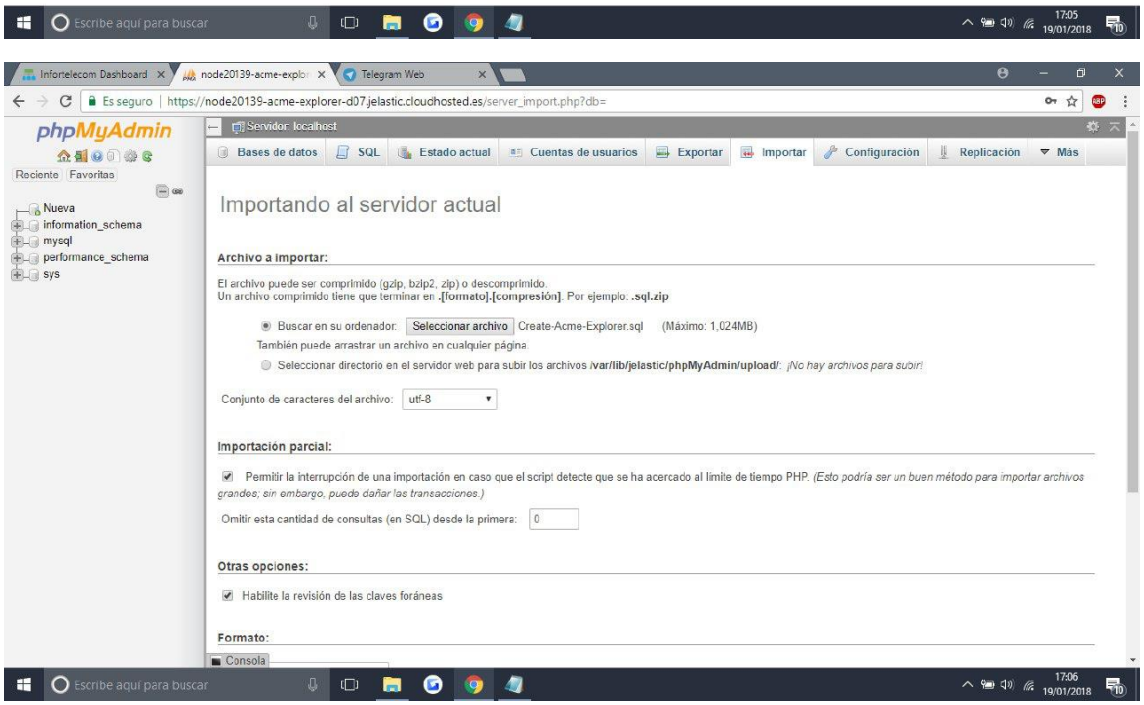


One of these files is called my.cnf, and is located at the root of our deployed project. It belongs to MySQL, and it includes many configuration options for it, such as port redirection and buffer size. However, as depicted below, we must change only one line to set the *lower_case_table_names* variable to 1. This makes table names case insensitive, which is useful because they are created in lowercase and we access them in uppercase from code.

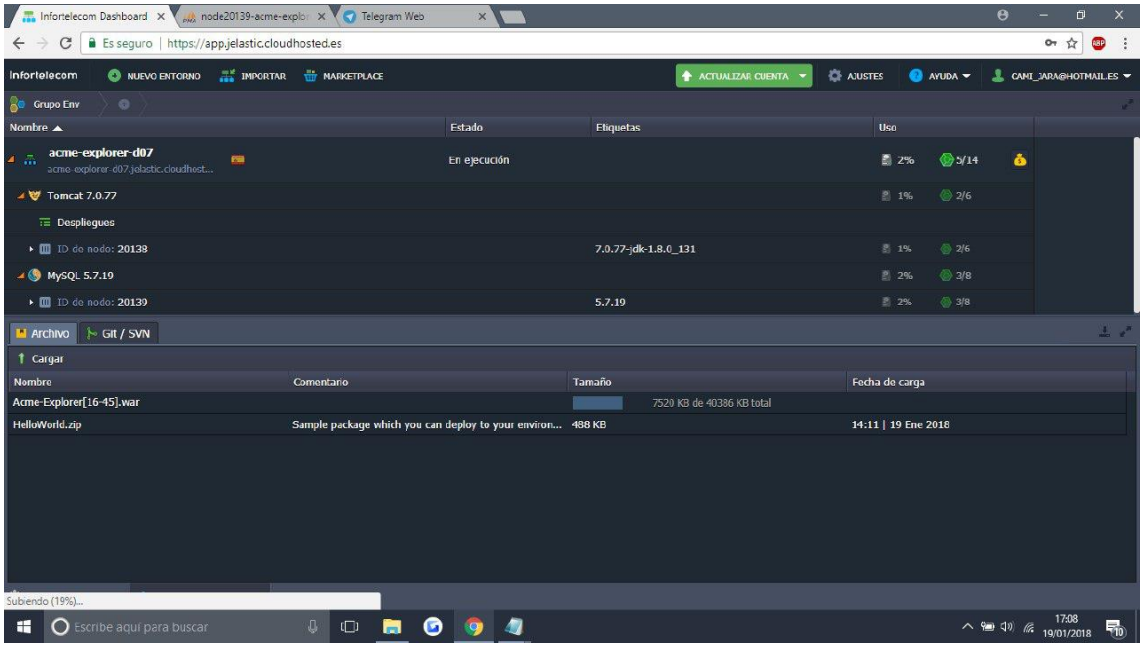


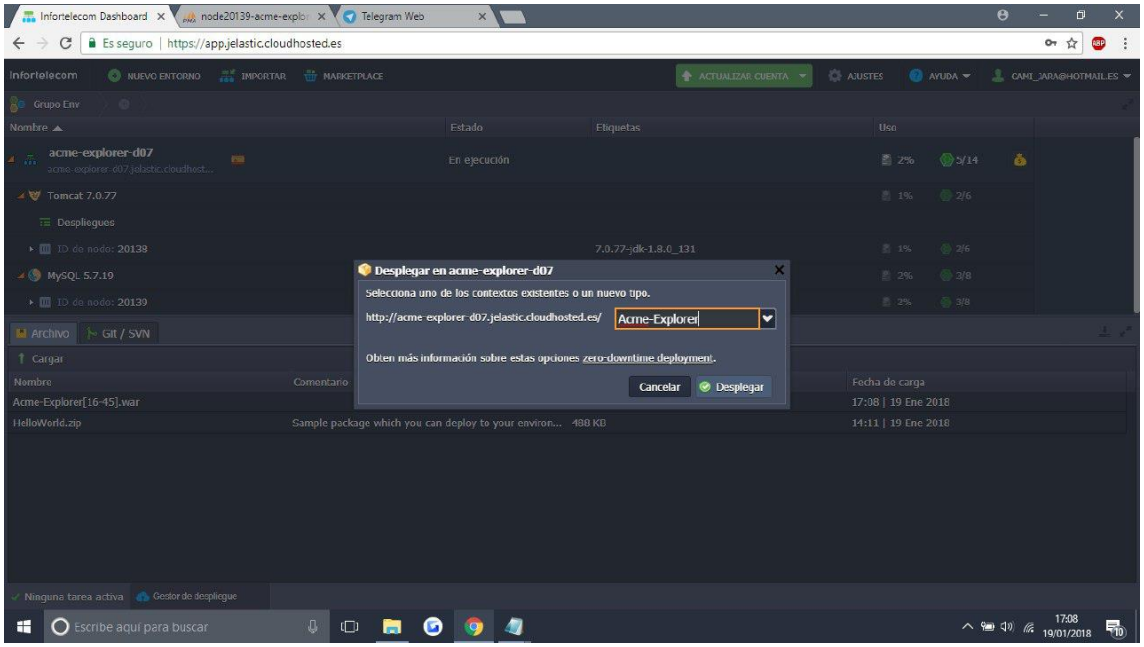
Once this is done, we must access the PHPMyAdmin control panel with username *root* and a password we should have received in an email when the server was created. Once in this control panel, we navigate to the *Import* tab and select our .sql file with all the necessary code to set up the Acme-Explorer database. **This is the one that was used in the pre-production configuration**, not the one used in development.





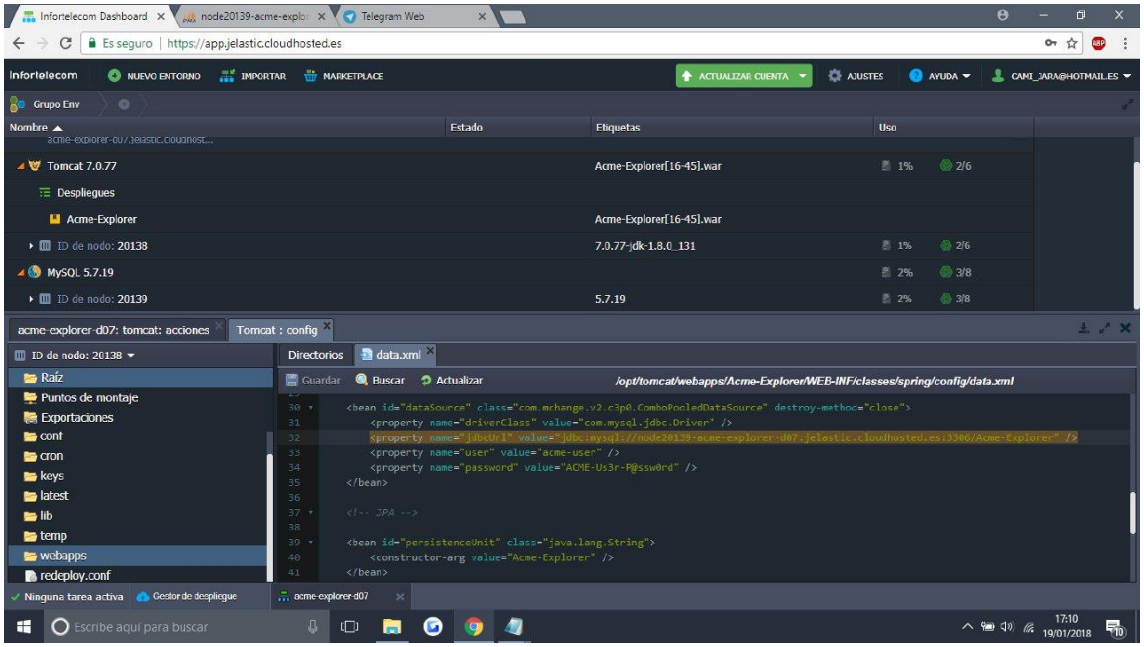
Once this is done, we will be shown a confirmation which includes the number of queries executed (in Acme-Explorer, it should be around 500; ours was 523), and then return to the Jelastic dashboard. In here, we use the deployment wizard to upload our WAR file, which should take no longer than a couple of minutes, and once it is uploaded a prompt should pop up asking a context in which to deploy (in our case, we have deployed to /Acme-Explorer).



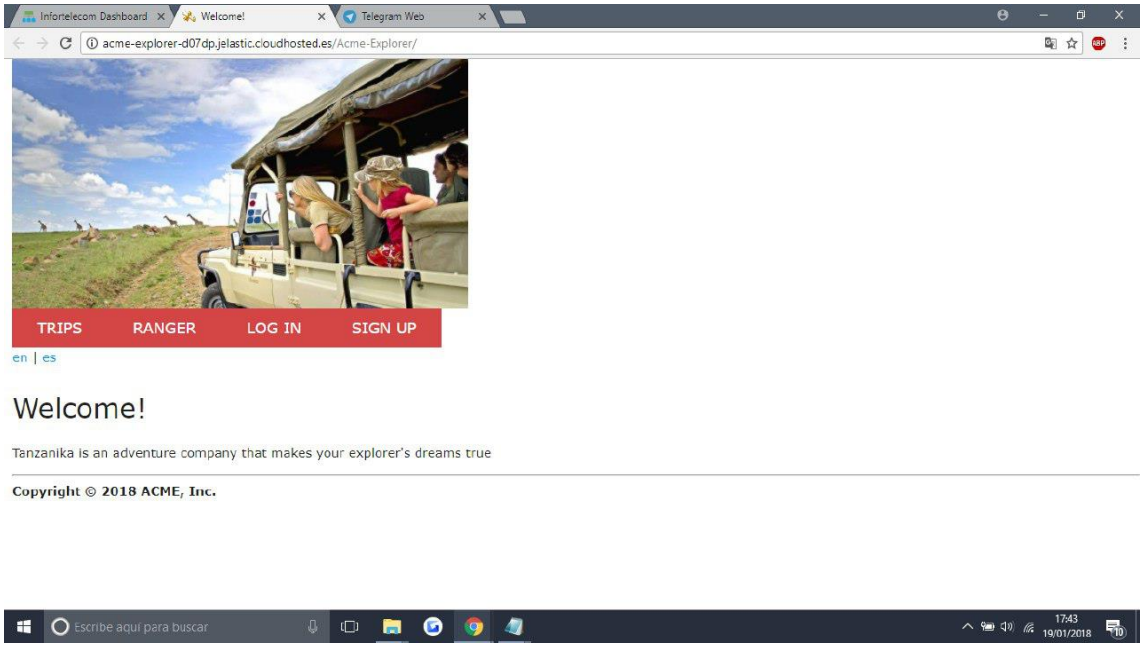


Step 3: Configuring our project to work on the cloud

Once the server is deployed, we must navigate to our project's *data.xml* file (the path is in the screenshot), where we must change the *jdbcUrl* variable (that is, the link to the database) to the one provided by our Jelastic-provided MySQL server. This should ensure a reliable connection to the database. At this point, if it doesn't do it itself, it is a good idea to restart the Tomcat node, to make sure all changes have applied. The MySQL node can also be restarted for extra caution, but it is not required.



Once this is done, we should have our Acme-Explorer up and running under our Jelastic-provided cloud account. To access it, we must only use the URL that appears in our dashboard and we will be organising trips in no time at all.



You can access our deployed app at http://acme-explorer-d07dp.jelastic.cloudhosted.es/Acme-Explorer/ and our Jelastic dashboard with username *cami_jara@hotmail.es* and password *despliegueJelasticd07*