# ICS 3104

# Lab 4

Now we need to create an API. We add features in our program to create the API, then we expose it. We will then create a second system which we will use to consume our system.

## Features:

* A user can order food from another system
* We can check the status of the order from another system

Proceed as follows:

Run the following queries in your btc3205 database

[CREATE](http://localhost:8080/phpmyadmin/url.php?url=http://dev.mysql.com/doc/refman/5.5/en/create-table.html) [TABLE](http://localhost:8080/phpmyadmin/url.php?url=http://dev.mysql.com/doc/refman/5.5/en/create-table.html) api\_keys(

id [INT](http://localhost:8080/phpmyadmin/url.php?url=http://dev.mysql.com/doc/refman/5.5/en/numeric-types.html) AUTO\_INCREMENT PRIMARY KEY,

user\_id [int](http://localhost:8080/phpmyadmin/url.php?url=http://dev.mysql.com/doc/refman/5.5/en/numeric-types.html),

api\_key [varchar](http://localhost:8080/phpmyadmin/url.php?url=http://dev.mysql.com/doc/refman/5.5/en/string-types.html)(255) [NOT](http://localhost:8080/phpmyadmin/url.php?url=http://dev.mysql.com/doc/refman/5.5/en/logical-operators.html#operator_not) NULL,

FOREIGN KEY (user\_id) REFERENCES [user](http://localhost:8080/phpmyadmin/url.php?url=http://dev.mysql.com/doc/refman/5.5/en/information-functions.html#function_user) (id))

For orders

[CREATE](http://localhost:8080/phpmyadmin/url.php?url=http://dev.mysql.com/doc/refman/5.5/en/create-table.html) [TABLE](http://localhost:8080/phpmyadmin/url.php?url=http://dev.mysql.com/doc/refman/5.5/en/create-table.html) orders(

order\_id [int](http://localhost:8080/phpmyadmin/url.php?url=http://dev.mysql.com/doc/refman/5.5/en/numeric-types.html) AUTO\_INCREMENT PRIMARY KEY,

order\_name [varchar](http://localhost:8080/phpmyadmin/url.php?url=http://dev.mysql.com/doc/refman/5.5/en/string-types.html)(255) [NOT](http://localhost:8080/phpmyadmin/url.php?url=http://dev.mysql.com/doc/refman/5.5/en/logical-operators.html#operator_not) NULL,

units [int](http://localhost:8080/phpmyadmin/url.php?url=http://dev.mysql.com/doc/refman/5.5/en/numeric-types.html),

unit\_price [double](http://localhost:8080/phpmyadmin/url.php?url=http://dev.mysql.com/doc/refman/5.5/en/numeric-types.html)(3,2),

orer\_status [varchar](http://localhost:8080/phpmyadmin/url.php?url=http://dev.mysql.com/doc/refman/5.5/en/string-types.html)(32))

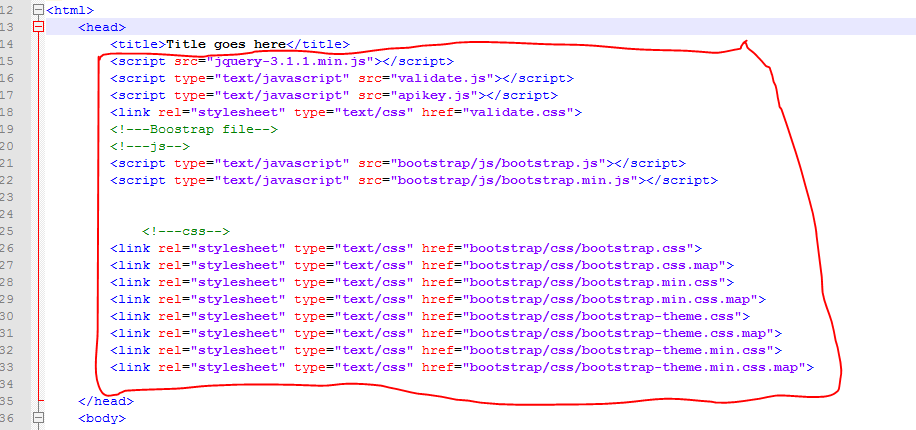
We need no database for the new database.

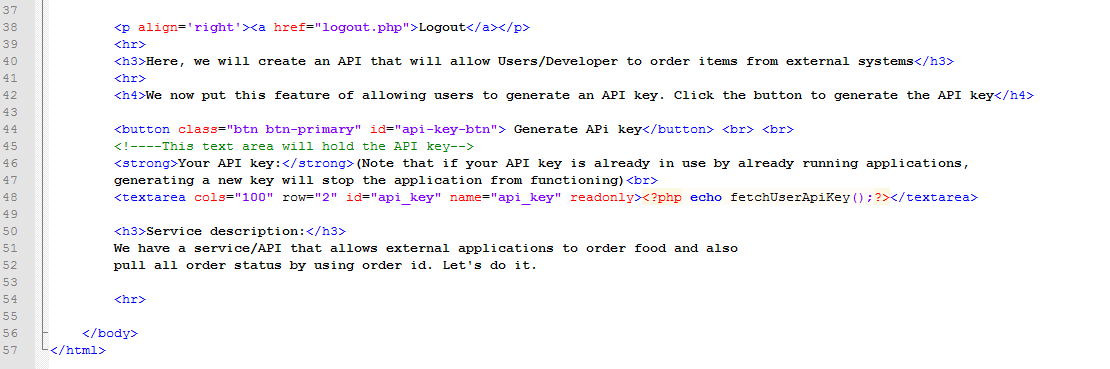
We now create an api path as shown



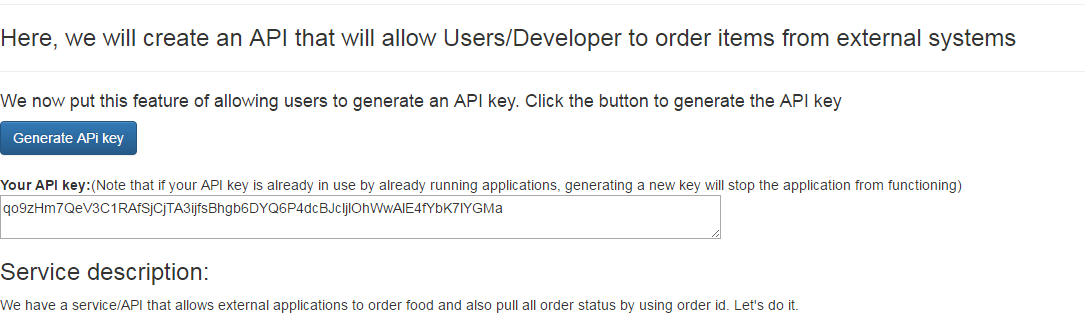
The API PHP folders will be placed in the orders folder.

Add the following codes in the private\_page.php file



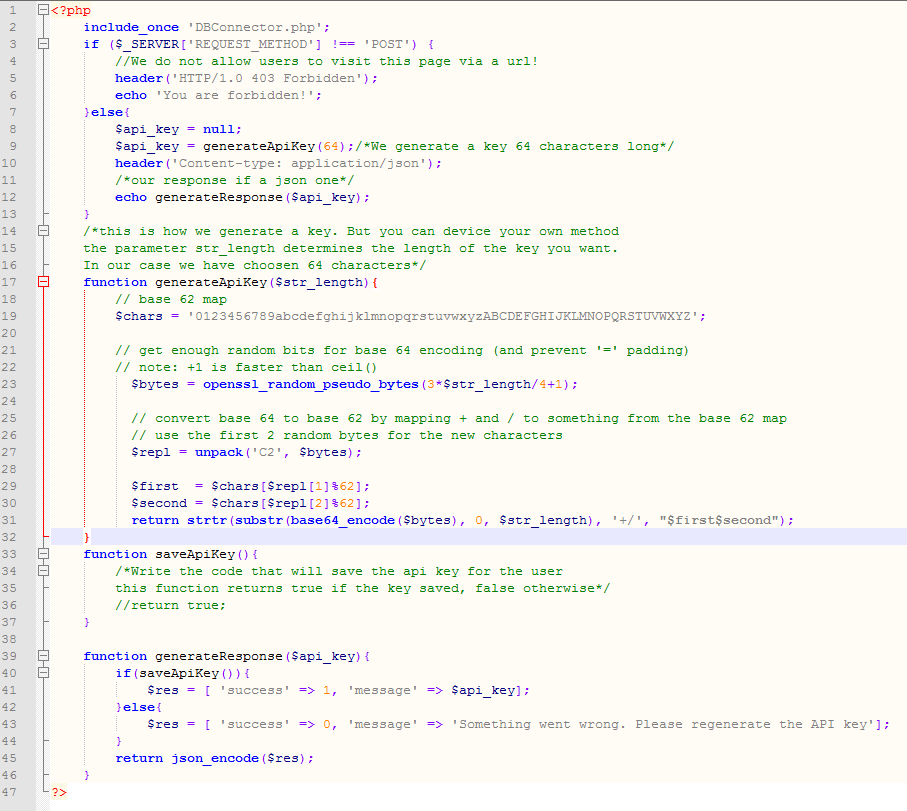


After this your interface should look this way:



The Generate API button is used to generate the API key when clicked. The generated API key is placed in the placed in the text area. Notice the id attribute value for the button (api-key-btn) and text area (api\_key)

Note that in the head section, I have included jQuery, bootstrap’s js and css. You can download the files or use it from any CDN network

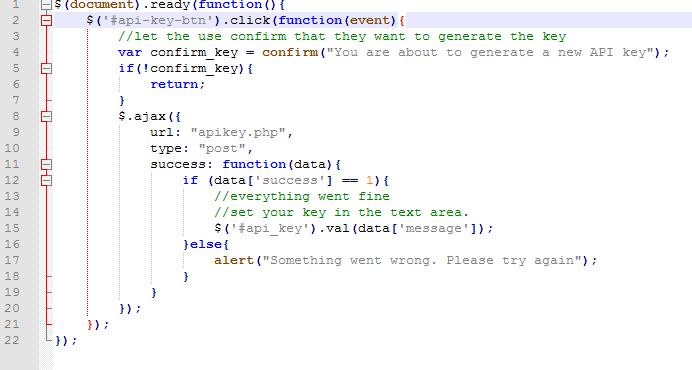
Now we write the PHP code to generate our API Key. We did not use a class, but you can choose to use a class.

Note how we generate the response json function generateResponse() (line39 - 46) .We first create an associative array, the we json encode using json\_encode() function

It is time now to do an ajax call using jquery. We click the button, and the key is generated.

What are doing is building an HTTP post with a URL we specify. The http request has no body, it has custom headers and it is of type post.

Our ***apikey.js*** looks like this



In your private\_page.php, do not forget to write a function called fetchUserApiKey(), with the code that fetches the key for the user currently logged in.

**-To be continued-**

**Now continue this wa………**

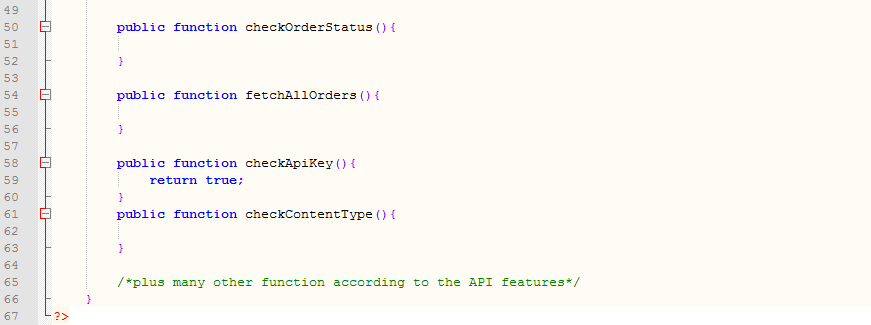
No that a user has generated a new API key, it is time to create the API features:

Inside the API path 

Create the following files: index.php and apiHandler.php (this has a class)

apiHandler.php

The index.php receives a request from and external and the apiHandler.php has the class that should carry the main application logic.



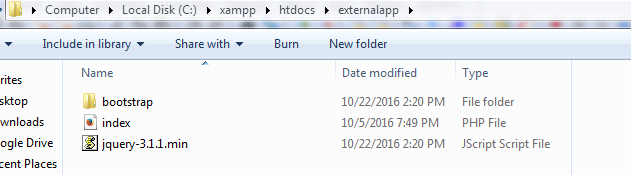
Then the index.php looks like this

Line 58-60 has a function, which you have to implement and check if the API key supplied is indeed in the database. If not, return false, otherwise return true.



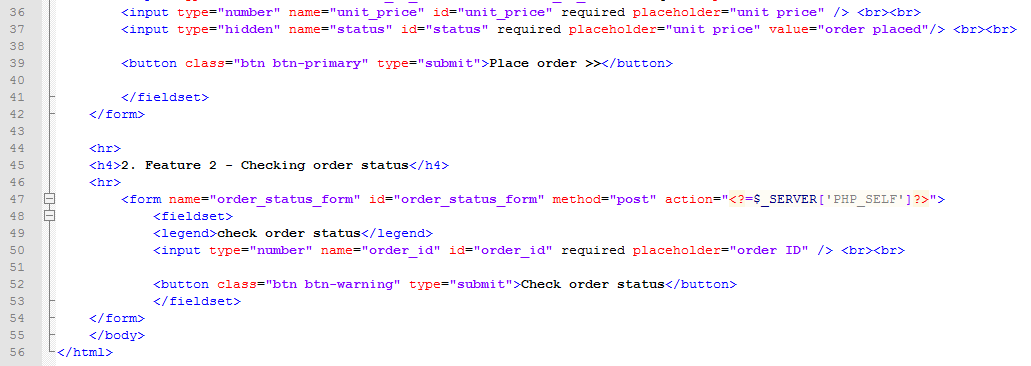
In line 9, we are receiving an API key passed to this application in the authorization header of the HTTP request. However, you have to receive all headers as shown in line 8.

Now we procced and create a new application in our server in a folder named externalapp



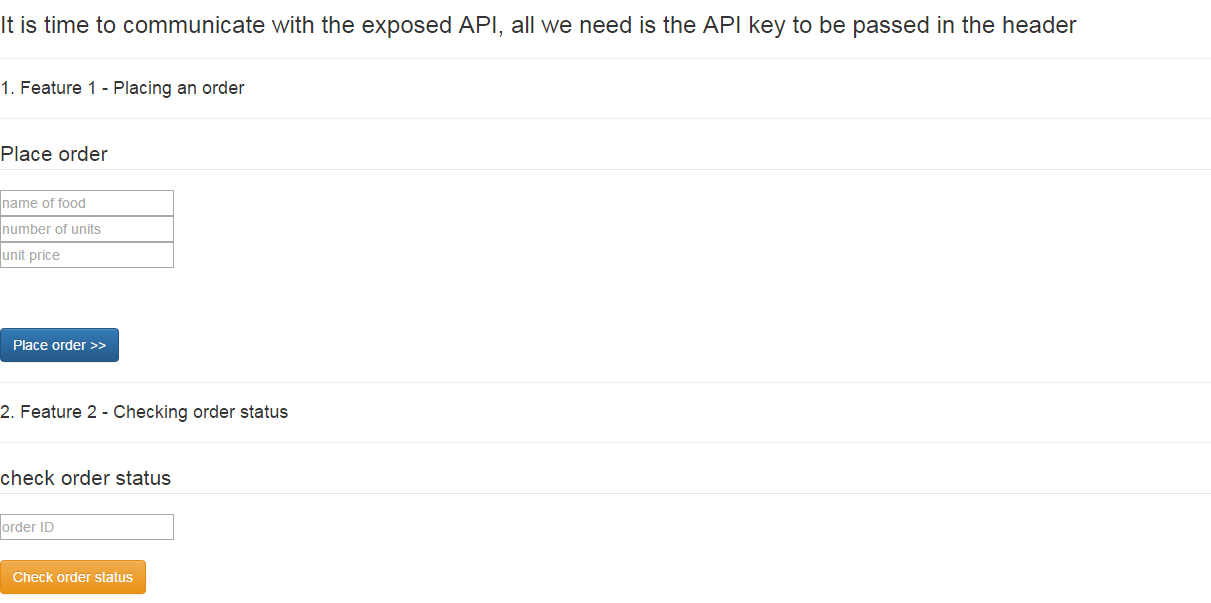
Notice how we have included the bootstrap and jQuery.

Now create a PHP file by name index to have the following contents

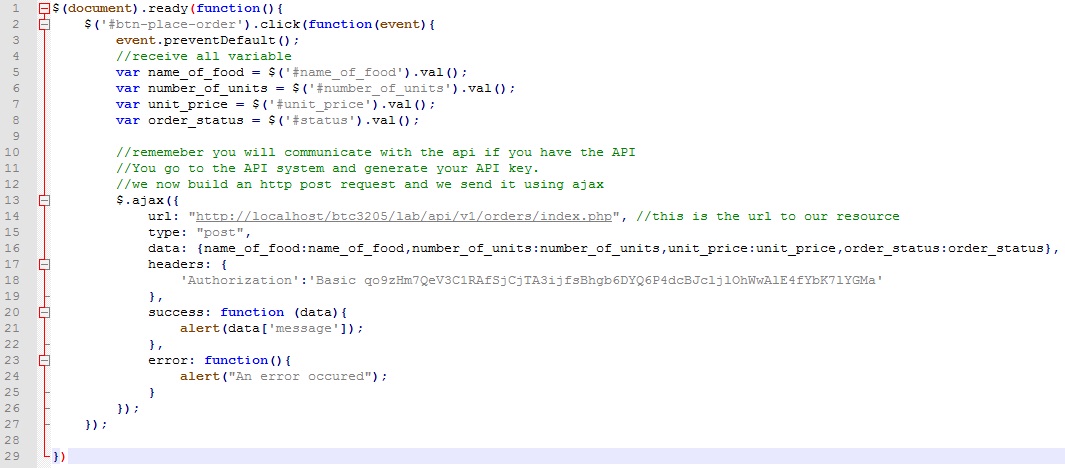


Notice that we have included a js file called placeorder.js in line 8.

Your interface should look like as shown below



We now proceed to do the content in placeorder.js.



Take note of the following:

Resource URL at line 14

API passed as an authentication header at line 17-19

Type is post

Ajax post Error handled at line 23-25

**Your work**

Now check your order status when the ‘check order status’ is clicked. You can add your code in the same JS file after line 27. The main logic should be put in the function at line 50 of apiHander.php file.