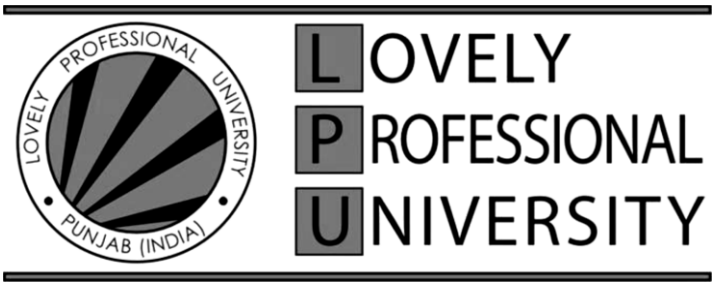

INT301 COURSE PROJECT REPORT

On

(OPEN SOURSE TECHNOLOGY: HTML, CSS, MYSQL, PHP)

https://github.com/Udipto-Goswami/International-Gems-Council

PROJECT CODE:

PROJECT TITLE: International Gems Council (IGC)

Submitted by

(Udipto Goswami)

udiptog@gmail.com

Registration No: 11401209

Programme Name: BTECH-MTECH (Dual) CSE

Under the Guidance of

Name of the Faculty: Shivali Chopra

School of Computer Science & Engineering

Lovely Professional University, Phagwara

(August-November, 2017)

INTRODUCTION

This project report provides an insight to the 3 months semester project developed regarding the course of INT301, open sources technologies. International Gems council (IGC) is a website which is developed entirely using the technologies used in web development. This project is a portrait of how a website works when both front-end and backend are coded, how the collaboration between various component adds up to form a single efficient system. The following are the technologies which are active components of the project of International gems Council:

- Ubuntu Linux (platform)
- HTML5
- CSS 3
- Bootstrap 3
- Xampp
- Maria DB (MySQL)
- PHP 5.7
- Apache 2.0
- Bootstrap Studio

International Gems Council (IGC) are leaders in Gems Testing (Like Diamonds, Ruby, Emerald, Yellow Topaz, Aqua Marine, Sapphire, and Pearl etc.) especially in Northern Region. Our Head Office is located in Shree G Mall, Karol Bagh, New Delhi (INDIA). IGC provides its technical services to gems and jewellery Exporters, Retailer Jewellers as well as general public. A Big Team of Expert Gemmologists, Graders, Instructors are there to provide education facilities also. Best part is, all reports have online access.

Considering the project as a proof of concept of what learnt throughout the semester, the content provided might be sufficient. However if to be seen for commercial purpose, there still might be need of more work and editing before making it online. The field which are needed only are retrieved while making ant transaction with the database. The LIMIT clause is also used many times in order to limit the search results. PHP function like header (“string”); is used to redirect from one page to another. For making code reusable, following is used, <?php include “string”; ?> This allowed to include files like header, database, footer etc. in different pages.

(IGC) COMPARISION

The existing systems which deals with keeping records on the Gems selling internationally are as follows:

https://gjepc.org/

This website has the database of the internationally selling gems all over the world. It is probably the database which have records of all the gems existing.

The approach of the international gems council is similar to of this website. But is differs on the presentation and the content the database holds. The gjepc.org only have a limited amount of data regarding any gem which is in the database. However IGC has the whole descriptions with it. It shows the history and the significance of a particular gem. Also it shows the basic information like the colour composition of the gem, weight, gross weight etc. Any gem, for identifying it uniquely are assigned with a unique id with it. Based on which the searches in the IGC is made. Also search is a working concept in IGC which is not quite unto the mark as it should be.

INNOVATION

There is no significant innovation of algorithms or any functionality which is portrait here, but the processing of images and storage could be considered an innovative way. The following section will explain how the image storage I designed.

Image Storage and Retrieval:

Traditionally MySQL gives us the privileges of storing the image files as a blob or a long blob file. Using this data-type we could store the image data in bit into the database directly.

But in many circumstances, the storage of image files in this way tends of shed significant load into the database due to which the query processes and the time taken might be compromised. In many cases, if the image size is larger than the allowed size, the image won't be accepted at all.

So in order to solve this problem, a mechanism is followed. Instead of storing my image as a blob file, I'm only storing the name of the image. Further in a specific folder my images are getting stored. This process makes the storage and retrieval very easy since we are only moving our image from the local computer to the folder present in the server. The image retrieval can easily be done using the absolute path of the image and the name.

LAYOUT

The layout of the whole project is based on Content management system (CMS). A CMS layout is very easy and simple way of organizing the data within folders. Mostly popular blog spots like WordPress are using this like of layout.

In this project There exists an includes folder which contains components like header.php and footer.php. This files are the examples of how code reusability in PHP is implemented. Every page is using these components which are described only once.

Similar to these are some more files like database.php containing the database connectivity variables. The nav-bar is made generic so that all page could use it.

Apart from the includes folder there is the admin folder where the access is only granted if the user is logged in as admin. Over there are the administrator files, showing the user database, gem database and options like created a gem record or new user etc.

Another folder called assets is there which actually contains all the CSS, JavaScript and images in it. Instead of scattering all the images here and there the images are kept at a fixed folder which is the assets folder. Further any fonts or glyphicon are also inserted there.

All the CSS file including the bootstrap and the custom user one are present in the CSS folder inside the assets folder. When the header wants to access the sources, it only have to give the relative path of the asset folder.

PROJECT INSIGHT

DATABASE:

The database of IGC is named as one_project and contains 2 tables. One of user and another of product. The users table is responsible for login and signup purposes and product table is of the gem records.

The following are the queries majorly used of these tables:

For logging in:

"SELECT username,password FROM users WHERE username = '$username' ";

Showing recent users:

"SELECT username, created,modified FROM users ORDER BY created DESC Limit 2";

Showing recent Gems records:

"SELECT product_no,weight,color_stwt FROM jproducts ORDER BY date_created DESC LIMIT 3";

For showing number of users:

"SELECT COUNT(*)AS total from users";

For showing number of Gem records:

"SELECT COUNT(*)AS total from jproducts";

For updating a record:

"UPDATE jproducts SETproduct_no = '{$product_number}',description ='{$description}',$query.= " weight = '{$weight}',gross_weight = '{$gross_weight}', diamond_weight = '{$diamond_weight}', color_stwt = '{$color_stwt}', WHERE id = '{$p_id}'";

For deleting a record:

"DELETE FROM jproducts WHERE product_no = '{$product_no}' ";

DATABASE CONNECTIVITY CODE:

```
<?php ob_start();
$db['db_host'] = "localhost";
$db['db_user'] = "root";
$db['db_pass'] = "";
$db['db_name'] = "oneproject";
```

```
    $connection=mysqli_connect($db['db_host'],$db['db_user'],$db['db_pass'],$db['db_na
    me']);
    $query = "SET NAMES utf8";
    mysqli_query($connection,$query);
?>
```

VALIDATIONS:

There are only the front end validations which are imposed on the project, no such significant constraints are given into the database. Simple validations like:

- Username should be of characters
- Gem record fields must be filled.

Are imposed.

SECURITY:

As we know the security in online websites are the most important bits. Therefore it is necessary to pay attention to it. In this project use of hash, SHA512 and MD5 could be seen.

The following is the PHP code snippet:

```
$password = hash('sha512',(md5($password)));
$query = "SELECT * FROM users WHERE username = '{$username}' ";
```

The hash value is the one which gets stored in the database.

Also is order to prevent the SQL injections, the following snippets are used:

```
$username = trim($username);
$password = trim($password);
$username = mysqli_real_escape_string($connection, $username);
$password = mysqli_real_escape_string($connection, $password);
```

These function helps and cleans up the inputs before going into the database.