Database Project

TITLE – Online Jewellery Shop

TABLE OF CONTENTS

1 INITIAL PROPOSAL

Online Jewellery Shop is used to build an application program that helps people to find and buy the latest design of jewelry with different categories like Gold Silver, Diamond. It is useful in the way that it makes an easier way to buy products online.

Today most jewelry shop is useful for shopping site. The admin has lots of paperwork and they are using a desktop, spreadsheet-like MS Excel application to manage data in soft copy about user records. In this proposed jewelry System it will run in server and the user can handle whole the registration activities. It has facilities to generate various types of reports (like pdf, excel) which are required by the management during event organizing. This application maintains the centralized database so that any changes done at a location reflects immediately. This is an online tool so more than one user can log in to the system and use the tool simultaneously.

- This application aims to reduce the manual effort needed to manage
- transactions and historical data used in various gods' owns. Also, this application provides an
- interface for users to view the details about events.

2 DATA SOURCES

Here, I have collected the data from various sources. The main sources are primary or secondary data collection.

- Primary Collection – Primary research entails the use of immediate data in determining the survival of the market. The popular ways to collect primary data consist of surveys, interviews, and focus groups, which show the direct relationship between potential customers and the companies. However, we have not attempted to collect the primary data for the development of this online software system.
- Secondary Collection – For secondary data, books and websites are referred to acquire knowledge about the running of such a system and which all modules can be incorporated into such systems. It is a means to reprocess and reuse collected information as an indication for betterments of the service or product.

3 DATA STORAGE ALTERNATIVES

MySQL is an open-source RDBMS that relies on SQL for processing the data in the database. MySQL provides APIs for the languages like C, C++, Eiffel, JAVA, Perl, PHP, and Python. MySQL is most commonly used for web applications and embedded applications and has become a popular alternative to proprietary database systems because of its speed and reliability. MySQL can run on UNIX, Windows, and Mac OS.MySQL is an essential part of almost every open-source PHP application. Good examples for PHP/MySQL-based scripts are phpBB, os Commerce and Joomla. One of the most important things about using MySQL is to have a MySQL specialized host.MySQL is the most popular Open Source Relational SQL database management system. MySQL is one of the best RDBMS being used for developing web-based software applications.MySQL is an open-source relational database management system (RDBMS) based on Structured Query Language (SQL). MySQL runs on virtually all platforms, including Linux, UNIX, and Windows. Although it can be used in a wide range of applications, MySQL is most often associated with web-

based applications and online publishing and is an important component of an enterprise stack called LAMP. LAMP is a Web development platform that uses Linux as the operating system, Apache as the Webserver, MySQL as the relational database management system, and PHP as the object-oriented scripting language. (Sometimes Perl or Python is used instead of PHP.)

4 RELATIONAL DATABASE DESIGN

User

Column Name	Type	Size	Description
id	Int	10	Id of the user /P.K
Name	Varchar	50	Name of the user
Surname	Varchar	20	User surname
Username	Varchar	30	User UserName
Password	Varchar	10	User Password
Email	Varchar	10	User EmailId

Cart

Column name	Type	Size	Description
id	Int	10	In this the id of the item is filled / P.K
jewel_id	Varchar	120	Name of the product id
Qty	Int	10	Quantity of the item

Jewelry

Column Name	Type	Size	Description
Id	Int	10	Id / P.K
Prodname	Varchar2	50	Name of the product
Path	Varchar2	50	Path of the image of product
Category	Varchar		Category of the product
price	Decimal	10,2	Price of the product
Description	Varchar2	100	Details about the product
Type	Varchar2	50	Type of product
no views	Number		Number of views of the product
top sell	Varchar2	120	Top sell product
Path	Varchar2	200	Path of an image
grand_total	Decimal	10,2	Total amount payable

Main Menu

Column name	Type	Size	Description
mmenu_id	Int	50	Id of the main menu
mmenu_name	Varchar	50	Name of the menu

Submenu

Column name	Type	Size	Description
Id	Int	50	Id of the product
mmenu_id	Int	50	Main menu id of product
smenu_name	varchar2	100	Name of the product

5 DATA DEFINITION LANGUAGE (DDL) SCRIPTS

A data dictionary, or Metadata Repository, as defined in the IBM Dictionary of computing, is a "centralized repository of information about data such as meaning, relationships to other data, origin, usage, and format. The term can have one of several closely related meanings about databases and database management systems (DBMS):

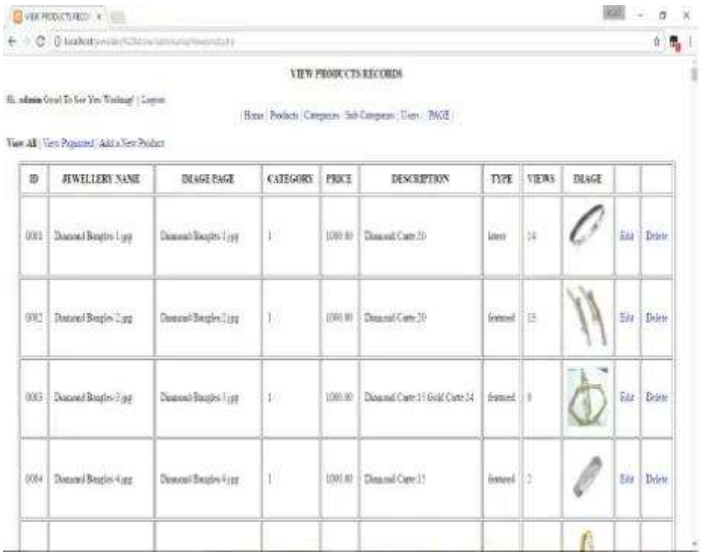
The terms data dictionary and data repository indicate a more general software utility than a catalog. A catalog is closely coupled with the DBMS software. It provides the information stored in it to the user and the DBA, but it is mainly accessed by the various software modules of the DBMS itself, such as DDL and DML compilers, the query optimizer, the transaction processor, report generators, and the constraint enforcer. On the other hand, a data dictionary is a data structure that stores metadata, i.e., (structured) data about information. The software package for a stand-alone data dictionary or data repository may interact with the software modules of the DBMS, but it is mainly used by the designers, users, and administrators of a computer system for information resource management. These systems maintain information on system hardware and software configuration, documentation, application, and users as well as other information relevant to system administration.

If a data dictionary system is used only by the designers, users, and administrators and not by the DBMS Software, it is called a passive data dictionary. Otherwise, it is called inactive data dictionary or data dictionary. When a passive data dictionary is updated, it is done so manually and independently from any changes to a DBMS (database) structure. With an active data dictionary, the dictionary is updated firsthand changes occur in the DBMS automatically as a result

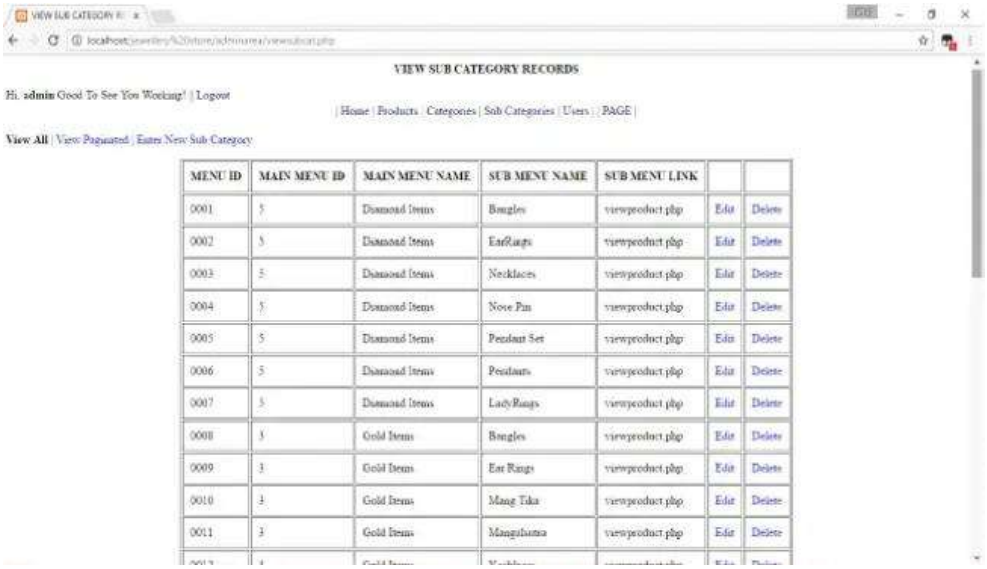
Name	Description
id	Identification Of User, Admin, and Jewellery Id of the product
email	Mail Address of User
mmenu_id	Main menu id of the product
Mmenu_name	Menu name of the category of the product
Submenu_name	Name of the product link to the main menu
Checkout	Check for the buy the product
Checked on	Date and time of buy the product by user

6 VIEWS

Product Information



Product Category



7 Database Security

An Online Jewellery Shop The main goal of this project was to create a shopping cart, which allows customers to shop and purchase Jewellery products online. Moreover, the project is also designed in such a way it lets managers manage the products information. Customers can order products, and they will be contacted to further process the orders.

In today's busy world, people don't have time for their personal needs. And the technology fasts that anyone can do by sitting in a room. If someone buys new things, he can buy them online with the help of the Internet.

The application carries two main components:

- Admin and
- Customer side

Admin side consists of the features such as Creating Username & Password, Input Items, Modify Items, Delete items, Query Sale Data, Query Database's data, and Logout.

The customer side consists of the features such as Select Products, Search Products, Buy Items, Continue Shopping, View Cart, Checkout, Sign-in, Creating an Account, Bill/Ship Information, Confirm, Send Order, and Delete Order. There are also future works for this application. There are mainly three such objectives which are as follows:

- To shop in the comfort of your home, without having to step out of the door.
- To be able to easily save money and compare prices from website to website.

8 LOCKING AND CONCURRENT ACCESS

Use the LOCK TABLE statement to lock one or more tables, table partitions, or table subpartitions in a specified mode. This lock manually overrides automatic locking and permits or denies access to a table or view by other users for the duration of your operation. Lock-based protocols help to eliminate the concurrency problem in DBMS for simultaneous transactions by locking or isolating a particular transaction to a single user. A lock is a data variable that is associated with a data item. This lock signifies that operations that can be performed on the data item

9 BACKING UP YOUR DATABASE

To prevent data loss, you should create backup copies of the database on an external storage device. During the security process, the database records the undertaken changes in a separate area and adds them to the relevant file at the end of the process.

cPanel is a popular control panel used by many web hosts. The backup feature can be used to backup your MySQL database. Do not generate a full backup, as these are strictly for archival purposes and cannot be restored via cPanel. Look for 'Download a MySQL Database Backup' and click the name of the database. A *.gz file will be downloaded to your local drive.

Quick backup process – When you backup all tables in the WordPress database without compression, you can use the simple method. To restore this backup, your new database should not have any tables.

- Log into phpMyAdmin on your server
- From the left side window, select your WordPress database. In this example, the name of the database is "wp".
- The right-side window will show you all the tables inside your WordPress database. Click the 'Export' tab on the top set of tabs.
- Ensure that the Quick option is selected, and click 'Go' and you should be prompted for a file to download. Save the file to your computer. Depending on the database size, this may take a few moments.

Commands:

- To back up all database tables

```
mysqldump --add-drop-table -h mysql_hostserver -u mysql_username -p
mysql_databasename
```

- To back up only certain tables from the database

```
mysqldump --add-drop-table -h mysql_hostserver -u mysql_username -p
mysql_databasename
```

10 PYTHON PROGRAMMING

Here are lists of coding for several sections:

- for the homepage in Online Jewelry Shop System

```
<!DOCTYPE html>
<html lang="en">
  <head>
    <meta charset="utf-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <title>Myrel Jewellery</title>
    {% load staticfiles %}
    <link href="{% static 'shop/bootstrap/css/bootstrap.min.css' %}" rel="stylesheet">
    <link href="{% static 'shop/bootstrap/css/style.css' %}" rel="stylesheet">
  </head>
  <body style="background-color: green">
    <!--  navbar starts -->
    <nav class="" style="border-radius:0px !important;margin-bottom:0px; background-color: black;">
      <div class="container-fluid">
        <!-- Brand and toggle get grouped for better mobile display -->
        <div class="navbar-header">
          <button type="button" class="navbar-toggle collapsed" data-toggle="collapse" data-target="#bs-example-navbar-collapse-1"
            <span class="sr-only">Toggle navigation</span>
            <span class="icon-bar"></span>
            <span class="icon-bar"></span>
            <span class="icon-bar"></span>
          </button>
          <a class="navbar-brand" href="{% url 'shop:home' %}"><span id="lione" style="color: red;">Myrel Jewellery Shop</span></a
        </div>

        <!-- Collect the nav links, forms, and other content for toggling -->
        <div class="collapse navbar-collapse" id="bs-example-navbar-collapse-1">
            <ul class="nav navbar-nav navbar-center">
              <li><a href="{% url 'shop:home' %}"><span id="" style="color: red">Home </span><span class="sr-only">(current)</span
              <li><a href="{% url 'shop:home' %}"><span id="lione" style="color: red">Rings & Neklace </span></a></li>

              <li><a href="{% url 'shop:about' %}"><span id="lione" style="color: red">About </span></a></li>
              {% if user.username %} <li><a href="{% url 'shop:logout' %}"><span id="lione" style="color: red">Logout {{user.usern
              <li><a href="{% url 'shop:cart' %}"><span class="glyphicon glyphicon-shopping-cart" style="color:red;"></span></a></
            </ul>
```

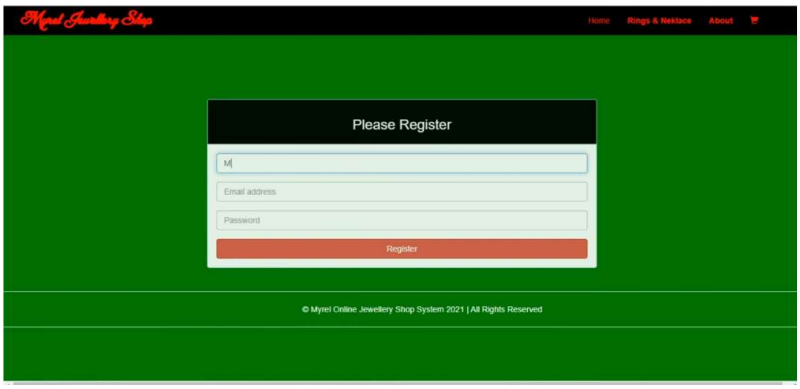
- for the cart in Online Jewelry Shop System

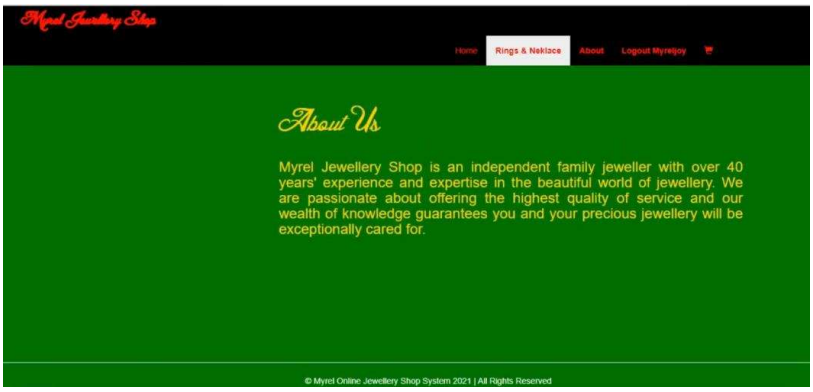
```
{% extends 'shop/base.html' %}

{% block body %}

<div class="container" style="padding-top:70px;">
  <div class="row">
    <div class="col-md-12 col-sm-12 col-xs-12">
      <div class="panel panel-info">
        <div class="panel-heading">
          <div class="panel-title">
            <div class="row">
              <div class="col-md-6">
                <h4><span class="glyphicon glyphicon-shopping-cart"></span> Cart</h4>
              </div>
              <div class="col-md-6">
                <a href="{% url 'shop:home' %}" class="btn btn-primary pull-right">Continue Shopping</a>
              </div>
            </div>
          </div>
        </div>
        <div class="panel-body">
          {% for item in buyList %}
            {% if item.user.username == user.username %}
            <div class="row">
              <div class="col-md-2 col-sm-12">
                <img src="{{ item.product.p_photo.url }}" alt="photo" width="20%"/>
              </div>
              <div class="col-md-6 col-sm-12">
                <h3><strong>{{ item.product.p_name }}</strong></h3>
                <!-- <h4><small>{{ item.product.p_detail }}</small></h4> -->
                <h4>Price : <strong>₹{% widthratio item.quantity 1 item.product.p_cost %}</strong></h4>
                <div class="dropdown">
                  <button class="btn btn-default dropdown-toggle" type="button" id="dropdownMenu1" data-toggle="dropdown" aria-has
                    Quantity
                    <span class="caret"></span>
```

- for the checkout in Online Jewelry Shop System

```
{% block body %}
<div class="container" style="padding-top:70px;">
<div class="row">
<div class="col-md-8 col-sm-8 col-xs-12">
<div class="panel panel-primary">
  <div class="panel-heading">Shopping Address</div>
  <div class="panel-body">
<form action="{% url 'shop:checkout' %}" method="post" enctype="multipart/form-data">
         {% csrf_token %}
           {% for field in form %}
           <div class="form-group">
             <input type="text" class="form-control" name="{{field.html_name}}" placeholder="{{field.label}}">
           </div>
           {% endfor %}
         <button type="submit" class="btn btn-success" style="width:100%;">Proceed</button>
       </form>
  </div>
</div>
</div>
<div class="col-md-4 col-sm-4 col-xs-12">
<div class="panel panel-success">
  <div class="panel-heading">
<span class="pull-right"><a href="{% url 'shop:cart' %}">Edit List</a></span>
Review Order
</div>
  <div class="panel-body">
{% for buy in buyItems %}
{% if buy.is_purchased == True and buy.user == user %}
    <h4>{{ buy.product.p_name }}</h4>
<h5>₱{% widthratio buy.quantity 1 buy.product.p_cost %}</h5>
<hr>
{% endif %}
{% endfor %}
  </div>
</div>
```

Screenshots:





11 SUGGESTED FUTURE WORK

The scope of the project includes that what all future enhancements can be done in this system to make it more feasible to us:

- Databases for different products ranges and storage can be provided.
- Multilingual support can be provided so that it can be understandable by the person of any language.
- More graphics can be added to make it more user-friendly and understandable.
- Manage & backup versions of documents online.

12 ACTIVITY LOG

S. NO.	***Activity***	***Description***
1.	Proposing title of the Project Along with a small description	1 – Week
2.	Data Collection Sources And Relational Database Design	2 – Week
3.	Indexes, Views, Triggers, Transactions	3 – Week, 4 – Week, 5 – Week
4.	Database Security, Backing Up the database	6 – Week, 7 – Week
5.	Python Programming And Php programming	8 – Week