DEVELOPMENT OF INVENTORY MANAGEMENT SYSTEM FOR ESTABLISHMENTS IN REPUBLIC OF INDIA

GRADUATE QUALIFICATION WORK
SUSU–02.04.2019.13-137-2598.GQW

Supervisor
Cand. Sci., Assoc. Prof.
___________ O.N. Ivanova

Author,
the student of the group CE-229
___________ M.S. Chaubey

Normative control
___________ O.N. Ivanova
“___”___________ 2019
Chelyabinsk–2019
MINISTRY OF SCIENCE AND HIGHER EDUCATION OF THE RUSSIAN FEDERATION

Federal State Autonomous Educational Institution of High Education
South Ural State University (National Research University)
School of Electronical Engineering and Computer Science
Department of Computer Science

APPROVED
Head of the department,
Dr. Sci., Prof.

__________ L.B. Sokolinsky
“___”___________ 2019

TASK
of the master graduate qualification work
for the student of the group KE-129
Mani Shanker Chaubey
in master direction 02.04.02
“Fundamental Informatics and Information Technologies”
(master program “Database Technologies”)

1. The topic (approved by the order of the rector from 25.04.2019 No. 899)
Development of inventory management system for establishments in Republic of India.

2. The deadline for the completion of the work: 05.06.2019.

3. The source data for the work

4. The list of the development issues
4.1. To make the comparative analysis of the existing software for inventories in different countries.
4.2. To study and compare the modern tools of web sites developing.
4.3. To implement and test the system for cloud based inventory system including working with all necessary information about personnel; management
of daily sales; management of products order, calculation of highest sales of a day or month.

5. **Issuance date of the task:** 09.02.2019.

**Supervisor**

Cand. Sci., Assoc. Prof. O.N. Ivanova

**The task is taken to perform** M.S. Chaubey
# TABLE OF CONTENTS

INTRODUCTION.................................................................................................................. 6  
1. THE ANALYSIS OF THE SUBJECT AREA................................................................. 9  
   1.1. The problem statement....................................................................................... 9  
   1.2. The used development tools.......................................................................... 10  
2. DEVELOPMENT OF A SITE.................................................................................... 13  
   2.1. Functional requirements................................................................................. 13  
   2.2. Use case diagram......................................................................................... 14  
3. IMPLEMENTATION OF THE WEB-APPLICATION.............................................. 21  
   3.1. Full implementation of the part of Administrator........................................... 21  
   3.2. Full implementation of the part of Special User and partial implementation  
        of the part of Ordinal User............................................................................. 26  
   3.3. File structure of the project.......................................................................... 27  
   3.4. PHP code screens for the project................................................................. 28  
4. TESTING OF THE WEB APPLICATION.................................................................. 36  
CONCLUSION.................................................................................................................. 38  
LITERATURE.................................................................................................................... 39
INTRODUCTION

Actuality

Inventory management is an important aspect of any successful business. It is the process of overseeing and controlling the flow of inventory units a business uses in the production or manufacture of goods for sale or distribution. Inventories are usually made up of a combination of goods, raw materials and finished products, and effective management of these items is essential to ensure optimal stock levels and to maximize the earning potential of the company. It also allows a business to prevent or mitigate any inventory-associated losses. Inventory management software is used by businesses for various reasons: it can track the costs of inventory throughout the manufacture and sales process, tell businesses when to replenish stock, and allow them to track profits. It can also be used to forecast inventory levels and prices, as well as expected product demand.

Effective inventory management is important as not only is inventory one of the most valuable assets to a business; there is a direct link between inventory levels and company profits. Inventory represents an investment that is tied up until either the item is sold, or it is used in the production of another item that is sold. Businesses are reliant on having items in stock; otherwise customers will simply go to a competitor who can provide what they want.

However, holding inventory in stock is not without costs – storage, insurance and maintenance all must be considered. When it comes to replenishing stock levels, most management plans seek to strike a balance between having enough units when required, and ensuring supplies are not overstocked. This is why having an inventory management system can be advantageous.

An inventory management system monitors all aspects of a company’s inventory as items move through the production and sales process. The process involves tracking customer orders, shipping, costs, stock and sales. Whether or
not a business has some form of inventory software in place, there are some critical elements every system needs in order to function efficiently.

**Research goal and objectives**

The goal of the research is the development of a system for cloud based inventory management to implement it at the various different business entities in the republic of India.

For the reaching this goal we must solve the following objectives:

1) to describe the subject domain;

2) to study the modern tools of web sites developing and make the comparative analysis of the content management frameworks for web-development;

3) to develop the structure of the required cloud based database for Inventory System;

4) to develop and implement the system for cloud based inventory system for the business entities in the Republic of India;

5) to test the system.

**The practical significance**

This project is useful for most of the business entities in the Republic of India because they all have the very similar algorithm of Inventory system. The difference between them is only about the product domain they use.

This project can be useful because it contains several features to assist the special experts in a company who calculates and checks the total sales and gives the possibility for usual workers to place an order if stock is out of inventory:

1) secure access for the private data about the users of web portal, including the individual identification at the site for all users;

2) saving the time and efforts while working at site;

3) ensuring data integrity among the different users of web portal;

4) uploading the media files related to the product with its description;
5) easily process of calculation of sale for a day and generate report for a month or a day between two calendar date;

6) easily dealing with the site for its friendly-man design;

7) ability for the following development of the site.

**Structure of the thesis**

The thesis consists of four chapters, introduction, conclusion and reference list.

In the first chapter, problem statement is represented, as well as the overview and comparative analysis of popular content management frameworks for web-development is done. Also, we describe the chosen development tools here.

In chapter two, there is a description of functional requirements, use case diagram, database scheme and the design of the application.

In chapter three, we show several fragments of PHP-code for implementing the basic functionality of the system and the algorithm of the Inventory Management System and its implementation in the web-application.

Chapter Four is devoted to the testing of the application.

The thesis has 38 pages; the list of references contains 20 resources.
1. THE ANALYSIS OF THE SUBJECT AREA

1.1. The problem statement

Inventory systems provide companies with the ability to track the supplies and raw materials needed to build product to meet customer demand. The accuracy of the inventory system affects purchasing, planning and production departments in the organization. The planning department uses the inventory data to create a production schedule to meet customer demand. Accurate inventory records allow the purchasing department to make an accurate assessment of lead times for purchasing materials for production and to ensure the production department has the materials and supplies available to build customer product.

**Employee Errors**

Employee errors can cause inaccuracies in the inventory records, which can cause purchasing to fail to purchase materials or to acquire an excessive amount of inventory. The employees responsible for transacting materials or work orders must have the training necessary to update the inventory system accurately. In addition, employees responsible for inventory management, such as cycle counters and inventory management specialists, must receive training in the specific inventory system used in the organization.

**Stock Outs**

Stock outs are shortages in inventory that can result from inaccurate records or a poor forecasting in the inventory system. The purchasing department must have accurate trigger points that determine when to make material purchases. Stock outs can result in product delays to customers.

**Excess Inventory**

Excess inventory results in additional costs to the organization in storage costs and funds tied up in unused stock. When companies do not use inventory quickly after purchase, the business begins to lose money on the materials. In
the event any of the material is defective, it can take much longer for the organization to discover the problem.

**Misplaced Inventory**

An inventory system must not only store quantity information, but it must also provide details of its location in the building. Misplaced inventory results in wasted time as workers search for lost materials. Time delays can result in late deliveries to customers as well.

**Lack of Optimization**

The inventory system must collect sufficient data to assist purchasing and planning to forecast future supply needs. A well-optimized inventory system can provide the company with information on quantities used in production as well as scrap and waste information. This data can help purchasing determine an accurate inventory level for production materials.

1.2. The used development tools

A database is an integrated collection of logically related records or files consolidated into a common pool that provides data for one or more multiple uses.

One way of classifying databases involves the type of content, for example: bibliographic, full-text, numeric, and image. Other classification methods start from examining database models or database architectures.

The data in a database is organized according to a database model. The relational model is the most common [6]. To add, access, and process data stored in a computer database, the developer needs a database management system such as MySQL. Since computers are very good at handling large amounts of data, database management plays a central role in computing, as stand-alone utilities, or as parts of other applications.

MySQL is a relational database management system. A relational database stores data in separate tables rather than putting all the data in one big
storeroom. It adds speed and flexibility. Defined relations making it possible to combine data from several tables on request link the tables. The SQL part of MySQL stands for “Structured Query Language” - the most common standardized language used to access databases.

MySQL is an open source software. It means that it is possible for anyone to use and modify. Anybody can download MySQL from the Internet and use it without paying anything. Anybody so inclined can study the source code and change it to fit their needs.

MySQL is very fast, reliable, and easy to use. MySQL also has a very practical set of features developed in very close cooperation with our users. MySQL was originally developed to handle very large databases much faster than existing solutions and has been successfully used in highly demanding production environments for several years. MySQL today offers a rich and very useful set of functions. The connectivity, speed, and security make MySQL highly suited for accessing databases on the Internet.

MySQL is a client/server system that consists of a multi-threaded SQL server that supports different backends, several different client programs and libraries, administrative tools, and a programming interface. MySQL has a lot of contributed software available.

It is very likely that you will find that your favorite application/language already supports MySQL [13].

PhpMyAdmin is a free software tool written in PHP, intended to handle the administration of MySQL over the web. It supports a wide range of operations on MySQL, and other DBMS. Frequently used operations (managing databases, tables, columns, relations, indexes, users, permissions, etc.) can be performed via the user interface, while the developer will have the ability to directly execute any SQL statement.

PhpMyAdmin comes with a wide range of documentation. The developers can use a variety of support channels to get help. To ease usage to a wide range
of people, phpMyAdmin is translated into 72 languages and supports both LTR and RTL languages [14].

PHP stands for Hypertext Preprocessor (no, the acronym doesn't follow the name). It's an open source, server-side, scripting language used for the development of web applications. By scripting language, we mean a program that is script-based (lines of code) written for the automation of tasks.

What does open source mean? Think of a car manufacturer making the secret to its design models and technology innovations available to anyone interested. These design and technology details can be redistributed, modified, and adopted without the fear of any legal repercussions.

Web pages can be designed using HTML. With HTML, code execution is done on the user's browser (client-side). On the other hand, with PHP server-side scripting language, it's executed on the server before it gets to the web browser of the user. PHP can be embedded in HTML, and it's well suited for web development and the creation of dynamic web pages for web applications, e-commerce applications, and database applications. It's considered a friendly language with abilities to easily connect with MySQL, Oracle, and other databases.
2. DEVELOPMENT OF A SITE

2.1. Functional requirements

This application must contain the following basic functions:

1) authorize the users;
2) connect to the database;
3) sales management;
4) user management;
5) product management.

The requirement “Authorize the users” means to have an opportunity for users to log in and log out the site. It is a special type of protection from the unauthorized access. There are three types of users:

1) Administrator;
2) Special User;
3) General User.

**Administrator** Dash board view contains highest sales product, Latest sales, recently added product, on the admin navigation the user will able to manage users and their groups, can add categories for the product, can view stock details, can add and manage products, add and manage sales, can generate sales report by date and month.

**Special User** can add an inventory location, can add the discount for the product, and can upload a media file related to product into database.

**User** can check inventory location for a product item, can place an order for the product in case of product is out of order, can add and manage the sales.

The application implement the following technical requirements:

1) operating system: different versions of Windows, Linux;
2) browser: Internet Explorer (from version 5), Mozilla Fireworks (from version 4), Opera (from version 5);
3) server: PHP version 5.1.6 or newer, MySQL (4.1+).
2.2. Use case diagram

Unified Modeling Language (UML) enable IT professionals to model computer applications. UML provides a unified standard modeling notation that IT professionals had been wanting for years.

Using UML, they could now read and disseminate the system structure and design plans just as construction workers have been doing for years with blueprints of buildings.

A use case diagram is a type of UML diagrams. It illustrates a unit of functionality provided by the system.

The main purpose of the use case diagram is to help development teams to visualize the functional requirements of a system, including the relations between “actors” (human beings who will interact with the system) and essential processes (use cases) [1].

The result of developing the use case model is shown in the fig. 1.

There are three main actors: Administrator, Special user and User. They use different use cases to work with tables in the database. For example, they can see sales, enter and modify data in different tables.

There are main four use cases for the system “Add sales”, “Manage Sales”, “Add Products” and “Manage products”.

2.3. Development of the database

Databases today are essential to every business. In essence, a database is a collection of information that exists over a long period of time, often many years. In common parlance, the term database refers to a collection of data that is managed by a DBMS.
The DBMS is expected to allow users to create new databases and specify their schemas (logical structure of the data), give users the ability to query and modify the data [10].

A DBMS is a powerful tool for creating and managing large amounts of data efficiently and allowing it to persist over long periods of time, safely. These systems are among the most complex types of software available [4].

MySQL, launched in 1995, has become the most popular open source DBMS. Another root cause of MySQL’s popularity has been the ongoing success of phpMyAdmin (www.phpmyadmin.net), a well-established MySQL web-based interface. Therefore, many websites use MySQL as their back-end data repository [13].

We decided to specify the following tables of the database:

- the table for categories contains information about the product categories which save categories names;
  
- the table, which saves the discount applied to specified product it contains attributes like product id, percentage, from date, to date;
  
- the table, which saves the location for inventory items like product location, product name;
  
- the tables, which saves the media file information for specific product it contains media file name and its type;
  
- the table, which saves product details (i.e., “Name”, “Quantity”, “Buying Price”, ”Sailing Price” etc.);
  
- the tables, which saves the user information (i.e., “User Name”, “User Status”, “User Level”);
  
- the table, which saves user group information (i.e., “Group Name”, “Group Status”);
  
- the table, which saves dates and documents about the loans for employees;
- the tables, which saves information about the sales (for example, “product”, “qty”, “price” and date);

- the tables, which saves information about deductions for employees and their types (“not present”, “penalty”, “vacation”);

- the table, which define the new orders if product is out of stock.

The scheme of the database is shown in the fig. 2. It consists of 11 tables, described below.

Fig. 2. The scheme of the database

The table for categories contains information about the product categories, which save categories names as in fig. 3.

Fig. 3. Structure of the table “Categories”
The table, which saves the discount applied to specified product it contains 6 Fields as in the fig. 4.

![Table structure of "Discount"](image)

**Fig. 4. Structure of the table “Discount”**

The table, which saves the location for inventory items. It consists of 4 fields as in the fig. 5.

![Table structure of "Inventory Location"](image)

**Fig. 5. Structure of the table “Inventory Location”**

The tables, which save the media, file information for specific product. It consists of fields as in the fig. 6.

![Table structure of "Media"](image)

**Fig. 6. Structure of the table “Media”**

The table, which saves product details. It consists of 8 fields as in the fig. 7.
The tables “sales” save the names of product sales with the sales date and price, quantity. The structure of sales table is in the fig. 8.

The table “User Order” connects with products. It consists of 4 fields as in the fig. 9.

The table “users” saves information about the web portal users and it connected with the user group. It consists of 10 fields as in the fig. 10.
The table User Group is saved information for user’s specific group and it connected with the users. The structure of the table is in the fig. 11.
3. IMPLEMENTATION OF THE WEB-APPLICATION

3.1. Full implementation of the part of Administrator

The implementation of the proposed system is described below by showing its pictures.

Fig. 12 shows the login form for the Inventory System.

![Welcome](image)

Fig. 12. The main page

The user who can use our website must be one of the organization's users. At first, he must login with his or her user name and password. Since logged he or she can be on their dashboard according their user level, e.g. Admin Dashboard (fig. 13).

![Inventory Management System](image)

Fig. 13. The page for the Admin Dashboard

Fig. 14 shows the total number of users in the system, total number of product categories, products and sales. It also includes a report related to the sales with the highest sailing product, and the latest sales.
Fig. 14. Admin can activate or deactivate a user role using this form.

Fig. 15 shows the interface of adding a new user in “User Management” section.

Fig. 15. Add a new User

In section “User Management” Admin can activate, deactivate, add, and remove user groups, fig. 16–17.

Fig. 16. The page with user management for activation or deactivation
After login, Admin can view and add product categories, as in the fig. 18.

After login, Admin can view the product details, as in the fig. 19.
If Admin wants to add a new product, he or she can add it, as in fig. 20.

![Add new product](image1)

**Fig. 20. The page for Add new product**

Admin can add a picture of a new product, as in fig. 21.

![Images for products](image2)

**Fig. 21. The page “Images for products”**

Admin has an access to the section “Sales” (fig. 22). He can add a sale for a particular date, edit a record of a sale, and remove the information (fig. 23).

![Sale section](image3)

**Fig. 22. Sale section**
Admin has functions to view sales report for a month or on chosen daily basis, as in the fig. 24–26.
3.2. Full implementation of the part of Special User and partial implementation of the part of Ordinal User

The special user’s dashboard looks as in fig. 27.

Fig. 27. Special User Dashboard

Fig. 28 and 29 display the screen for Client Management.

Fig. 28. Add a new client
Sales management form is available for the Special User and the Ordinal User. The interface of it is the same as in Admin part.

3.3. File structure of the project

Fig. 30 demonstrates the content of included folder.

It contains files, which are used in all the project, for example, files related to database configuration, session handling file, SQL queries file and upload file which used for upload to files on server.

Fig. 31 demonstrates files inside the layout folder.

It contains a file for design separate menu based on the specific user groups, it also contains files for common header and footer related code, which is used across the application.

Fig. 32 demonstrates the folders and files inside the lib folder.

It further contains folder for CSS, Images and J Query library used in the application.
3.4. PHP code screens for the project

Fig. 33 and 34 show the screen for the code written to handle the Admin Menu layout of the pages. Figures 35 and 36 show the screen for user session handling. Fig. 37–39 show the screen for database queries used in the application.
Fig. 40 shows the function written for display monthly sale from the database and fig. 41 demonstrates the monthly sale function called by Monthly sales generation form. It takes one parameter for year.
Fig. 35. User session handling

```php
<?php
  session_start();
  class Session {
    public $this_user_is_logged_in = false;
    function __construct() {
      $this->flash_msg = null;
      $this->userLoginSetup();
    }
    public function isLoggedIn() {
      return $this->user_is_logged_in;
    }
    public function login($username, $password) {
      $SESSION['user_id'] = $user_id;
      $private function userLoginSetup() {
        if (isset($SESSION['user_id'])) {
          $this->user_is_logged_in = true;
        } else {
          $this->user_is_logged_in = false;
        }
    }
  }
  public function logout() {
    unset($SESSION['user_id']);
    public function msg($type = '', $msg = null) {
      if (empty($msg)) {
        if (isset($SESSION['msg'])) {
          $SESSION['msg'] = str_replace(array('?', '!', ',','@'), array('danger', 'info', 'warning', 'success'), $type);
        } $SESSION['msg'][$type] = $msg;
      } else {
        return $this->flash=$msg;
      }
    } private function flash_msg() {
      if (isset($SESSION['msg'])) {
        unset($SESSION['msg']);
        $SESSION['msg'] = $SESSION['msg'];
      } elseif (isset($SESSION['msg'])) {
        unset($SESSION['msg']);
      }
    } $session = new Session();
    $msg = $session->msg();
  }
?>
```

Fig. 36. User session logout code

```php
<?php
  require_once('includes/load.php');
  // Function for finding all database table rows by table name
  function find_all($table) {
    global $db;
    if (!is_array($table)) {
      return find_by_sql("SELECT * FROM ".db->escape($table));
    }
  } // Function for performing queries
  function find_by_sql($sql) {
    global $db;
    $result = $db->query($sql);
    $result->set = $db->fetchAll($result);
    return $result;
  } // Function for finding data from table by id
```
Fig. 38. Database handling file

```
/* Function for finding data from table by id */
function find_by_id($table, $id)
{
    global $db;
    $sql = (is_int($id) ?
        $sql = "SELECT * FROM "$table" WHERE id='$id';"
    ) ?
        $sql = "SELECT * FROM "$table" WHERE id='$id';"
    :
        $sql = "SELECT * FROM "$table" WHERE id='$id';"
    ;
    $result = $db->query($sql)
    return $result;
    else
        return null;
}
```

Fig. 39. Database handling using SQL queries

```
/* Function for counting data in a table */
function count_by_id($table)
{
    global $db;
    if(!$tableExists($table))
        return false;
    $sql = "SELECT COUNT(id) AS total FROM "$table" ;"
    $result = $db->query($sql)
    return($db->fetch_assoc($result));
}
```

Fig. 40. Monthly sale function

```
/* Function for finding data from table by id */
function find_by_id($table, $id)
{
    global $db;
    $sql = (is_int($id) ?
        $sql = "SELECT * FROM "$table" WHERE id='$id';"
    ) ?
        $sql = "SELECT * FROM "$table" WHERE id='$id';"
    :
        $sql = "SELECT * FROM "$table" WHERE id='$id';"
    ;
    $result = $db->query($sql)
    return $result;
    else
        return null;
}
```
Fig. 41. Monthly sales generation code

Fig. 42 shows the function written for daily sales report generation formed from the database and figure 43 demonstrates the call of daily sale function from the sql.php file, which should take two parameters (month and year).

```php
// Check if the user has permission to view this page
if ( ! page_has_level(2) )
    die;

$year = date("Y");
$year = month_sales($year);

// Include the header file
header("layout/head.php");

<!-- Monthly Sales -->

<!-- Table -->
<table class="table table-bordered table-striped">
    <th>Product name</th>
    <th>Product price</th>
    <th>Total sales</th>
</table>
```

Fig. 42. The daily sale function
Fig. 43. Daily sale.php called the daily sale function

Figure 44 shows the code screen for gaining total sales between any given dates by using a function, which finds sales by date.

This function will ask user to supply two arguments as start date and end date.

Figure 45 shows the code screen for the sale report process.php, which will call the function “find_sales_by_date” from the sql.php file and store the result into result variable.
Fig. 45. Sales report process.php calls the function find sales by id

Figures 46, 47 are the continuation of the sales report process.php to calculate the profit.

**Algorithms applied to calculate the total profit.**

To calculate the profit first we sum total buying price for the every sale and then subtract it from the total sailing price for every Sale.

\[
\text{Profit} = \text{total buying price} - \text{total sailing price}.
\]

In figure 47 it is shown the function of the total price calculation, which is further used by the sales report process to calculate the total profit based on the differences of buying and sailing price. Figure 36 shows how this function used to generate the profit.

Fig. 46. The function total price in function.php file
Fig. 47. Total price function used in sale process report.php to generate profit
4. TESTING OF THE WEB APPLICATION

Functional system tests should be based around coverage of the functionality described in the requirements, but it is common for the design document to be used as the baseline for testing because the requirements cannot be related to the end product. In the worst case, system tests can become large-scale repetitions of unit tests [5]. Each test of my system contains input and output information. Therefore, we compare the actual results and the expected results (table).

Table. The protocol of functional testing of the site

<table>
<thead>
<tr>
<th>No.</th>
<th>Function</th>
<th>Expected result</th>
<th>Obtained result</th>
<th>Conclusion</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.</td>
<td>To give Admin permission to see the Admin Dash Board</td>
<td>Admin can see the Admin Dash Board</td>
<td>Admin can see the Admin Dash Board</td>
<td>The function works</td>
</tr>
<tr>
<td>2.</td>
<td>To give all users the permission to see the login page at all sections</td>
<td>Any user can see the login page at all sections</td>
<td>Any user can see the login page at all sections</td>
<td>The function works</td>
</tr>
<tr>
<td>3.</td>
<td>To show all statics related to sales on admin dashboard</td>
<td>Admin can see all sales related statics on dashboard</td>
<td>Admin can see all sales related statics on dashboard</td>
<td>The function works</td>
</tr>
<tr>
<td>4.</td>
<td>To user management by Admin</td>
<td>The Admin can create new user by storing their basic details related to login like email ,username ,password to the user table in database</td>
<td>The Admin can create new user by storing their basic details related to login like email ,username , password to the user table in database</td>
<td>The function works</td>
</tr>
<tr>
<td>5.</td>
<td>Under the user management navigation admin can activate or deactivate user, update the password for user.</td>
<td>Admin can update user details</td>
<td>Admin can update user details</td>
<td>The function works</td>
</tr>
<tr>
<td>6.</td>
<td>Admin can manage groups</td>
<td>Admin can create a new group or update the group information</td>
<td>Admin can create a new group or update the group information</td>
<td>The function works</td>
</tr>
</tbody>
</table>

36
<table>
<thead>
<tr>
<th>No.</th>
<th>Function</th>
<th>Expected result</th>
<th>Obtained result</th>
<th>Conclusion</th>
</tr>
</thead>
<tbody>
<tr>
<td>7.</td>
<td>Under the Categories navigation admin can add new categories for product or can update it</td>
<td>Admin can create new categories or update to database table categories.</td>
<td>Admin can create new categories or update to database table categories.</td>
<td>The function works</td>
</tr>
<tr>
<td>8.</td>
<td>Under admin product navigation</td>
<td>Admin can add a new product to database table product.</td>
<td>Admin can add a new product to database table product.</td>
<td>The function works</td>
</tr>
<tr>
<td>9.</td>
<td>Under admin product navigation admin can view the listing of products.</td>
<td>The admin can views the list of product.</td>
<td>The admin can views the list of product.</td>
<td>The function works</td>
</tr>
<tr>
<td>10.</td>
<td>Under the Manage product section.</td>
<td>The admin can manage the products to the product table in database.</td>
<td>The admin can manage the products to the product table in database.</td>
<td>The function works</td>
</tr>
<tr>
<td>11.</td>
<td>Under the image section of admin navigation.</td>
<td>The administrator can upload an image of product to database table media.</td>
<td>The administrator can upload an image of product to database table media.</td>
<td>The function works</td>
</tr>
<tr>
<td>12.</td>
<td>Under the sales Report navigation</td>
<td>Admin can add new sales</td>
<td>Admin can add new sales</td>
<td>The function works</td>
</tr>
<tr>
<td>13.</td>
<td>Under the sales Report navigation</td>
<td>Admin can view the sales for a month</td>
<td>Admin can view the sales for a month</td>
<td>The function works</td>
</tr>
<tr>
<td>14.</td>
<td>Under the sales report navigation</td>
<td>Admin can view sales between two dates</td>
<td>Admin can view sales between two dates</td>
<td>The function works</td>
</tr>
</tbody>
</table>
CONCLUSION

The goal of the research was the development of a system for cloud based inventory management to implement it at the various different business entities in the republic of India.

Within reaching this goal, we solved the following objectives:

1) the modern tools of web sites developing and make the comparative analysis of the content management frameworks for web-development are studied;

2) the structure of the required cloud based database for Inventory System is developed;

3) the system for cloud based inventory system for the business entities in the Republic of India is designed and implemented;

4) the system is tested.
LITERATURE


