VISVESVARAYA TECHNOLOGICAL UNIVERSITY

"Jnana Sangama", Belagavi-590018

Database Management System

Mini Project Report

On

"WATER CAN MANAGEMENT SYSTEM"

Submitted in partial fulfillment of the requirement of V Semester Database Management System Laboratory

Submitted by,

PRAJJWAL KUMAR	**1DT19IS093**
PRIYANSHU SINGH	**1DT19IS101**

Under the guidance of

Ms. Vasudha G Rao
Asst. Professor
Dept. of ISE
DSATM, Bangalore.

DEPARTMENT OF INFORMATION SCIENCE AND ENGINEERING

DAYANANDA SAGAR ACADEMY OF TECHNOLOGY AND MANAGEMENT
(Affiliated to Visvesvaraya Technological University, Belagavi & Approved by AICTE, New Delhi)
All Engineering Branches Accredited by NBA, New Delhi
Opp. Art of Living, Udayapura, Kanakapura Road, Bangalore- 560082

2020-2021

DAYANANDA SAGAR ACADEMY OF TECHNOLOGY AND MANAGEMENT

(Affiliated to Visvesvaraya Technological University, Belagavi & Approved by AICTE, New Delhi)
All Engineering Branches Accredited by NBA, New Delhi
Opp. Art of Living, Udayapura, Kanakapura Road, Bangalore- 560082

DEPARTMENT OF INFORMATION SCIENCE & ENGINEERING

Certificate

This is to certify that the mini-project work entitled **"WATER CAN MANAGEMENT SYSTEM"** is carried out by **PRAJJWAL KUMAR(1DT19IS093) and PRIYANSHU SINGH(1DT19IS101)** in partial fulfillment for the requirement of V Semester DataBase Management System Laboratory(18CSL58) in **Information Science and Engineering** of the **Visvesvaraya Technological University, Belagavi** during the year 2021-2022. It is certified that all the corrections/suggestions indicated for the given internal assessment have been incorporated in the report. This report has been approved as it satisfies the academic requirements with respect to the mini-project work.

Signature of the Guide

Ms. Vasudha G Rao
Asst. Professor, Dept. of ISE
DSATM, Bangalore.

Signature of the HOD

Dr. Sumithra Devi K A
Prof. & Head, Dept. of ISE
DSATM, Bangalore.

External Viva

Name of the Examiners	**Signature with date**
1.	
2.	

ACKNOWLEDGEMENT

We express our profound gratitude to **Dr. M Ravishankar, Principal, DSATM, Bangalore,** for providing the necessary facilities and an ambient environment to work.

We are grateful to **Dr. Sumithra Devi K A, Dean Academics and Head of Department, Information Science and Engineering, DSATM, Bangalore,** for valuable suggestions and advice throughout our work period.

We would like to express our deepest gratitude and sincere thanks to our guide **Ms. Vasudha G Rao, Assistant Professor, Department of Information Science and Engineering, DSATM, Bangalore,** for teaching the course, keen interest and encouragement, whose guidance made the work more efficient.

We would like to thank all the staff members of **Department of Information Science and Engineering** for their support and encouragement during the course of this project.

Definitely most, we would like to thank our parents, all my family members and friends, without whose help and encouragement this creative assessment would have been impossible.

ABSTRACT

In today's world, there is a great problem with the supply of drinking water. This project presents a management proposal to improve the efficient use of water resources in water supply systems. This is based on management tools, project management and is organized into three levels of planning (strategic, tactical and operational), following definitions of theories of strategic planning. This paper details these levels of planning, with a focus on strategic management, i.e., action plans at the strategic level, describing a methodology and detailing the main tasks that should be executed, as well as the main tools that can be used in each task.

Water supply system of a city is in fact a major industry which produces, stores, and distributes water for humans. As such, the services provided by a managing entity to a community have fundamentally two objectives, which are Preservation of public health and social Purpose. With this approach, you cannot admit that the company pays additional costs arising from potential inefficiencies, whether source technical, commercial or managerial. So this is the primary reason for us choosing this project so that all the Management, Reports, Monthly Income and many different Filters can be obtained all at a same place. Thus providing easy and efficient way of Water Can Management.

TABLE OF CONTENTS

CHAPTER 1

INTRODUCTION

Water can management system is basically based on daily life problems especially in metropolitan areas. Our project is based on the supply of these cans to the customers

In earlier times, the river water was clean and clear due to which there is no issue for drinking water. However, with industrialization the river water started becoming polluted due to industrial waste. This gives rise to the concept of purification of water. First step in the methodology is the definition of the objectives, vision, mission and values of the water utilities. The strategic objectives for water utilities are the following (ISO 24512: 2007): (i) protection of public health; (ii) satisfaction of the needs and expectations of the users of the service; (iii) provision of service in normal and emergency conditions; (iv) sustainability of the water utilities; (v) promote the sustainable development of the community; and (vi) environmental protection.

In our web page, we are updating details in our admin page regarding all sales. All details/orders we are getting through calls or social media. We used to instruct our employers to supply according to the details filled in the admin page.

CHAPTER 2

Requirement Analysis

The requirement analysis specifies the requirements needed to develop a project. In this phase, we collect the requirements needed for designing the project. The requirements collected are then analyzed and carried to the next phase.

2.1 FRONT END TECH STACK

- **HTML**
- **CSS**
- **JavaScript**

2.2 BACKEND

For backend we have used PHP, MySql as database and apache using Xampp for the server in localhost.

- **PHP**
- **MYSQL**
- **XAMPP**

CHAPTER 3

DESIGN

3.1 ER DIAGRAM

An Entity – Relationship model (ER model) describes inter-related things of interest in a specific domain of knowledge. An ER model is composed of entity types (which classify the things of interest) and specifies relationships that can exist between instances of those entity types.

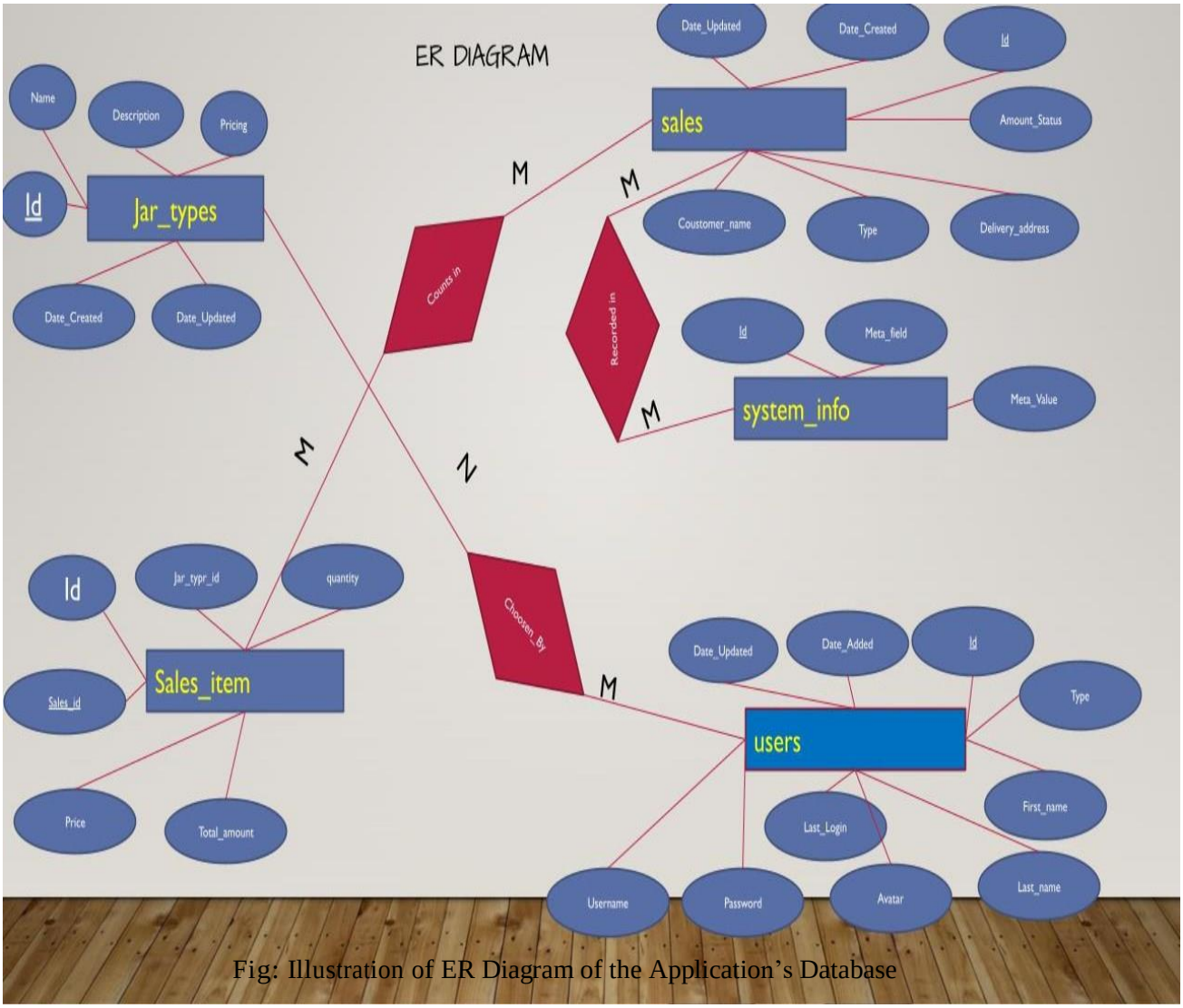


Fig: Illustration of ER Diagram of the Application's Database

3.2 SCHEMA DIAGRAM

The term "schema" refers to the organization of data as a blueprint of how the database is Constructed (divided into database tables in the case of relational databases).

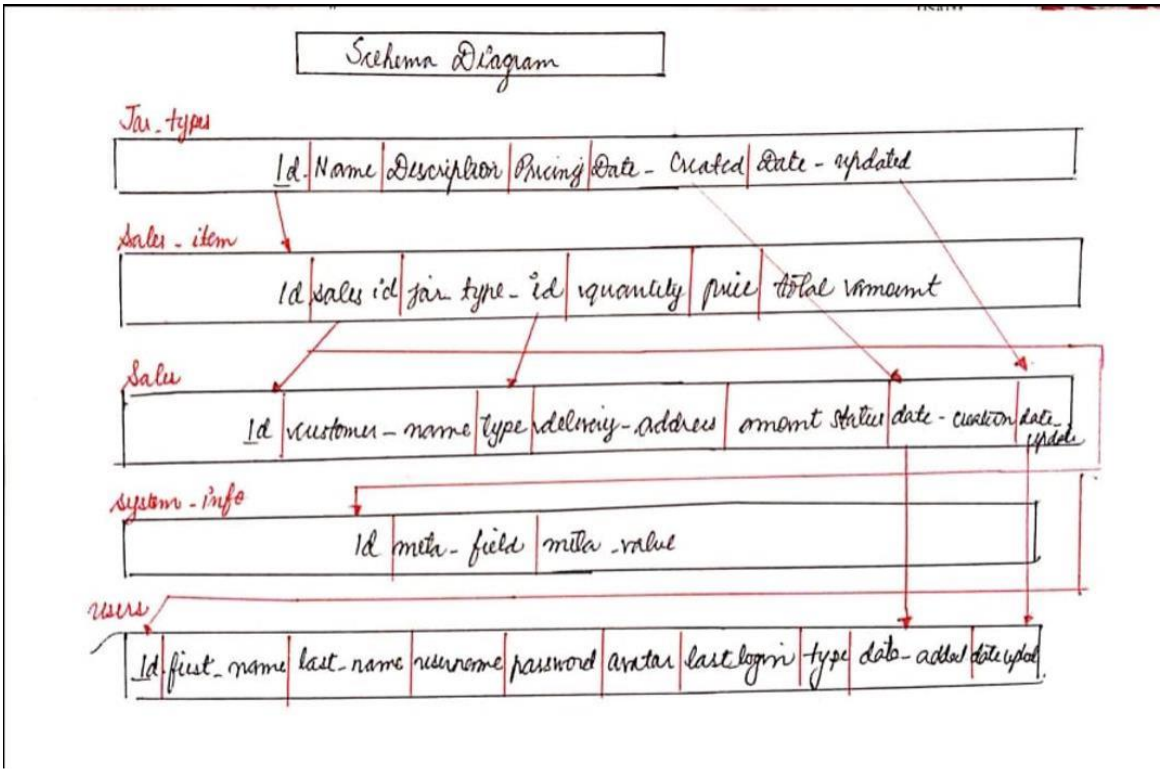


Fig: Figure showing Schema Diagram of the Application's Database

In this Schema diagram we have the Tables as jarTypes, sales, salesItems, systemInfo, Users.The attributes of table jar_types are id, name, description, pricing, date_created and date_updated. The attributes of table sales are id, customer_name, type, delivery_address , amount, status, date_created and date_updated.The attributes of table sales_items are id, sales_id, jar_type_id, quantity, price and total_amount. The attributes of table system_info are id, meta_field and meta_value.. " the attributes of table users are id, firstname, lastname, username, password, avatar, last_login, type, date_added and date_updated.

CHAPTER 4

IMPLEMENTATION

4.1 FRONTEND TOOLS

For frontend we have used HTML, CSS and JavaScript.

1. HTML

(HyperText Markup Language) is the most basic building block of the Web. It defines the meaning and structure of web content. Other technologies besides HTML are generally used to describe a web page's appearance/presentation (CSS) or functionality/behavior (JavaScript).

"Hypertext" refers to links that connect web pages to one another, either within a single website or between websites. Links are a fundamental aspect of the Web. By uploading content to the Internet and linking it to pages created by other people, you become an active participant in the World Wide Web.

HTML uses "markup" to annotate text, images, and other content for display in a Web browser. HTML markup includes special "elements" such as <head>, <title>

An HTML element is set off from other text in a document by "tags", which consist of the element name surrounded by "<" and ">". The name of an element inside a tag is case insensitive. That is, it can be written in uppercase, lowercase, or a mixture. For example, the <title> tag can be written as <Title>, <TITLE>, or in any other way.

2. CSS

Cascading Style Sheets (CSS) is a style sheet language used for describing the presentation of a document written in a markup language such as HTML.CSS is a cornerstone technology of the World Wide Web, alongside HTML and JavaScript.

CSS is designed to enable the separation of presentation and content, including layout, colors, and fonts .This separation can improve content accessibility; provide more flexibility and control in the specification of presentation characteristics; enable multiple web pages to share formatting by specifying the relevant CSS in a separate . Css file, which reduces complexity and repetition in the structural content; and enable the Css file to be cached to improve the page load speed between the pages that share the file and its formatting.

Separation of formatting and content also makes it feasible to present the same markup page in different styles for different rendering methods, such as on-screen, in print, by voice (via speech-based browser or screen reader), and on Braille-based tactile devices. CSS also has rules for alternate formatting if the content is accessed on a mobile device.

The name cascading comes from the specified priority scheme to determine which style rule applies if more than one rule matches a particular element. This cascading priority scheme is predictable.

The CSS specifications are maintained by the World Wide Web Consortium (W3C). Internet media type (MIME type) text/css is registered for use with CSS by RFC 2318 (March 1998). The W3C operates a free CSS validation service for CSS documents.

In addition to HTML, other markup languages support the use of CSS including XHTML, plain XML, SVG, and XUL.

3. JAVASCRIPT

often abbreviated JS, is a programming language that is one of the core technologies of the World Wide Web, alongside HTML and CSS. Over 97% of websites use JavaScript on the client side for web page behavior, often incorporating third-party libraries. All major web browsers have a dedicated JavaScript engine to execute the code on users devices.

JavaScript is a high-level, often just-in-time compiled language that conforms to the ECMAScript standard. It has dynamic typing, prototype-based object-orientation, and first-class functions. It is multi-paradigm, supporting event-driven, functional, and imperative programming styles. It has application programming interfaces (APIs) for working with text, dates, regular expressions, standard data structures, and the Document Object Model (DOM).

The ECMAScript standard does not include any input/output (I/O), such as networking, storage, or graphics facilities. In practice, the web browser or other runtime system provides JavaScript APIs for I/O.

JavaScript engines were originally used only in web browsers, but are now core components of some servers and a variety of applications like the most important Node.js.

Although Java and JavaScript are similar in name, syntax, and respective standard libraries, the two languages are distinct and differ greatly in design.

4.2 BACKEND TOOLS

For backend we have used PHP, MySql as database and apache using Xampp for the server in localhost.

1.PHP

is a general-purpose scripting language geared towards web development. It was originally created by Danish-Canadian programmer Rasmus Lerdorf in 1994. The PHP reference implementation is now produced by The PHP Group.PHP originally stood for Personal Home Page, but it now stands for the recursive initials PHP: Hypertext Preprocessor.

PHP code is usually processed on a web server by a PHP interpreter implemented as a module, a daemon or as a Common Gateway Interface (CGI) executable. On a web server, the result of the interpreted and executed PHP code – which may be any type of data, such as generated HTML or binary image data – would form the whole or part of an HTTP response. Various web template systems, web content management systems, and web frameworks exist which can be employed to orchestrate or facilitate the generation of that response. Additionally, PHP can be used for many programming tasks outside the web context, such as standalone graphical applications and robotic drone control. PHP code can also be directly executed from the command line.

The standard PHP interpreter, powered by the Zend Engine, is free software released under the PHP License. PHP has been widely ported and can be deployed on most web servers on a variety of operating systems and platforms.

The PHP language evolved without a written formal specification or standard until 2014, with the original implementation acting as the de facto standard which other implementations aimed to follow. Since 2014, work has gone on to create a formal PHP

2.MySQL

Is an open-source relational database management system (RDBMS). Its name is a combination of "My", the name of co-founder Michael Widenius's daughter,and "SQL", the abbreviation for Structured Query Language. A relational database organizes data into one or more data tables in which data types may be related to each other; these relations help structure the data. SQL is a language programmers use to create, modify and extract data from the relational database, as well

as control user access to the database.

MySQL was owned and sponsored by the Swedish company MySQL AB, which was bought by Sun Microsystems (now Oracle Corporation).In 2010, when Oracle acquired Sun, Widenius forked the open-source MySQL project to create Maria DB.

4.3 CONNECTIVITY TO THE DATABASE

Xampp using Apache is Web Server on Localhost on the same IP. For connectivity, we use XAMPP app so that we can get the localhost connection to the web browser.

In XAMPP, we start the APACHE and MYSQL modules

As a Web server, Apache is responsible for accepting directory (HTTP) requests from internet users and sending them their desired information in the form of files and web pages. Much of the Web's software and code is designed to work along with Apache's features.

MySQL is a relational database management system based on SQL - Structured Query Language. Here we are using this server in XAMPP for connecting our database frontend.

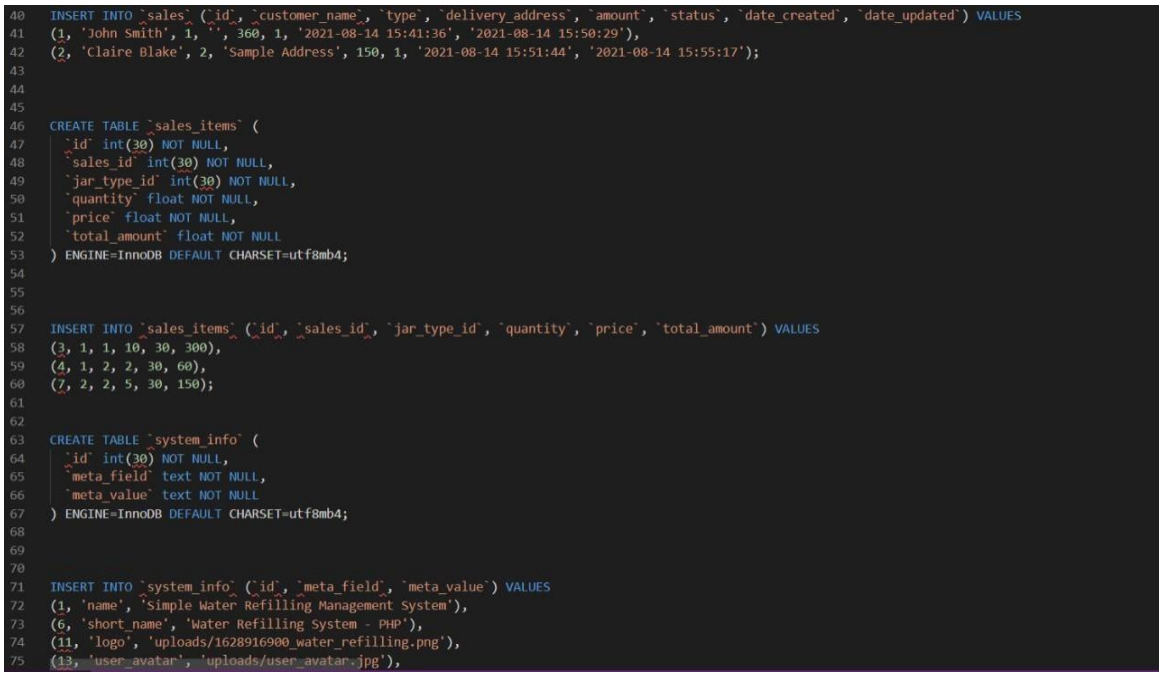
CHAPTER 5

SNAPSHOTS

The below four screenshots is code of the database of Water Can Management.

```
SET SQL_MODE = "NO_AUTO_VALUE_ON_ZERO";
START TRANSACTION;
SET time_zone = "+00:00";

CREATE TABLE `jar_types` (
  `id` int(30) NOT NULL,
  `name` text NOT NULL,
  `description` text NOT NULL,
  `pricing` float NOT NULL,
  `date_created` datetime NOT NULL DEFAULT current_timestamp(),
  `date_updated` datetime DEFAULT NULL ON UPDATE current_timestamp()
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4;

INSERT INTO `jar_types` (`id`, `name`, `description`, `pricing`, `date_created`, `date_updated`) VALUES
(1, 'Slim Container with cap and faucet', '<span style=\"color: rgb(0, 0, 0); font-family: \"Open Sans\", Arial, sans-serif; font-size: 14px; text-align: justify;\">Integ
(2, 'Round Container with Cap', '<p><span style=\"color: rgb(0, 0, 0); font-family: &quot;Open Sans&quot;, Arial, sans-serif; font-size: 14px; text-align: justify;\">Nunc

CREATE TABLE `sales` (
  `id` int(30) NOT NULL,
  `customer_name` text NOT NULL,
  `type` tinyint(4) NOT NULL DEFAULT 1 COMMENT '1 = walk-in, 2 = for delivery',
  `delivery_address` text NOT NULL,
  `amount` float NOT NULL,
  `status` tinyint(1) NOT NULL DEFAULT 0 COMMENT '0= Unpaid, 1=Paid',
  `date_created` datetime NOT NULL DEFAULT current_timestamp(),
  `date_updated` datetime DEFAULT NULL ON UPDATE current_timestamp()
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4;
```

Screenshot 1: The Screenshot contains Description of table “jar_types” & “sales” the attributes of table jar_types are id, name, description, pricing, date_created and date_updated. The attributes of table sales are id, customer_name, type, delivery_address , amount, status, date_created and date_updated.



```
INSERT INTO `sales` (`id`, `customer_name`, `type`, `delivery_address`, `amount`, `status`, `date_created`, `date_updated`) VALUES
(1, 'John Smith', 1, '', 360, 1, '2021-08-14 15:41:36', '2021-08-14 15:50:29'),
(2, 'Claire Blake', 2, 'Sample Address', 150, 1, '2021-08-14 15:51:44', '2021-08-14 15:55:17');

CREATE TABLE `sales_items` (
  `id` int(30) NOT NULL,
  `sales_id` int(30) NOT NULL,
  `jar_type_id` int(30) NOT NULL,
  `quantity` float NOT NULL,
  `price` float NOT NULL,
  `total_amount` float NOT NULL
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4;

INSERT INTO `sales_items` (`id`, `sales_id`, `jar_type_id`, `quantity`, `price`, `total_amount`) VALUES
(3, 1, 1, 10, 30, 300),
(4, 1, 2, 2, 30, 60),
(7, 2, 2, 5, 30, 150);

CREATE TABLE `system_info` (
  `id` int(30) NOT NULL,
  `meta_field` text NOT NULL,
  `meta_value` text NOT NULL
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4;

INSERT INTO `system_info` (`id`, `meta_field`, `meta_value`) VALUES
(1, 'name', 'Simple Water Refilling Management System'),
(6, 'short_name', 'Water Refilling System - PHP'),
(11, 'logo', 'uploads/1628916900_water_refilling.png'),
(13, 'user_avatar', 'uploads/user_avatar.jpg'),
```

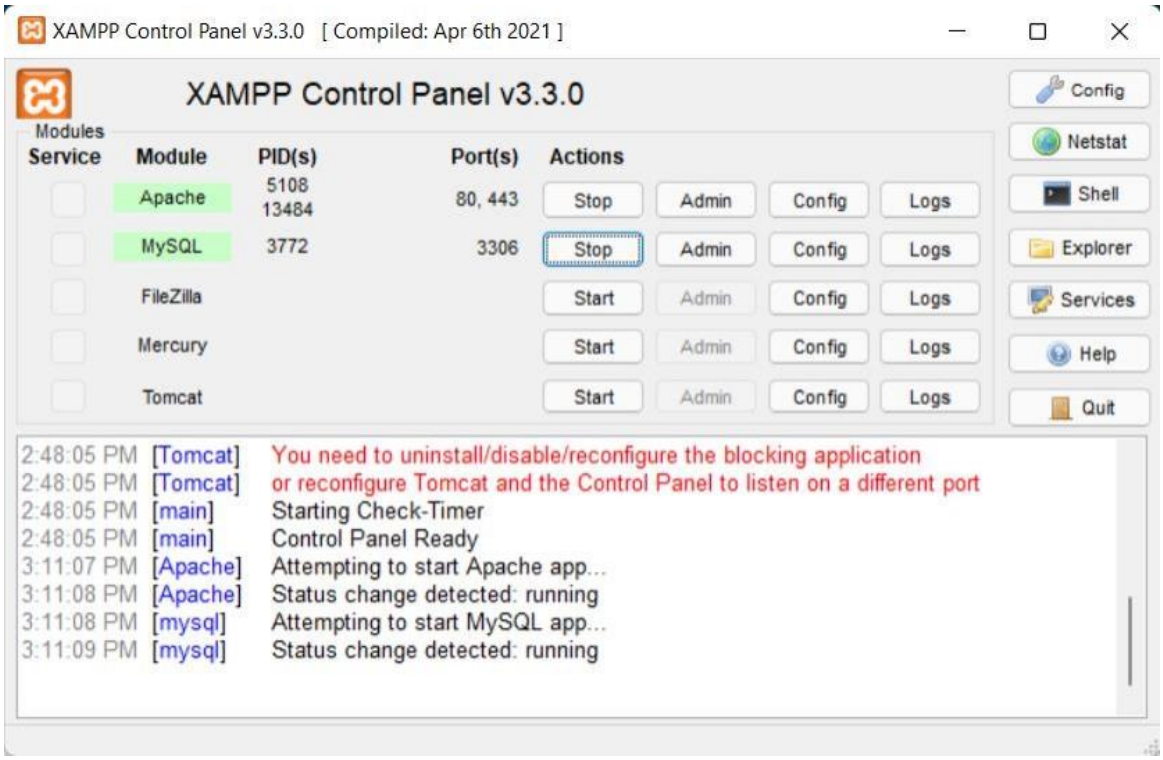
Screenshot 2: The Screenshot contains Description of table “sales_items” & “system_info” the attributes of table sales_items are id, sales_id, jar_type_id, quantity, price and total_amount. The attributes of table system_info are id, meta_field and meta_value.

```
(14, 'cover', 'uploads/1626249540_dark-bg.jpg');



CREATE TABLE `users` (
  `id` int(50) NOT NULL,
  `firstname` varchar(250) NOT NULL,
  `lastname` varchar(250) NOT NULL,
  `username` text NOT NULL,
  `password` text NOT NULL,
  `avatar` text DEFAULT NULL,
  `last_login` datetime DEFAULT NULL,
  `type` tinyint(1) NOT NULL DEFAULT 0,
  `date_added` datetime NOT NULL DEFAULT current_timestamp(),
  `date_updated` datetime DEFAULT NULL ON UPDATE current_timestamp()
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4;



INSERT INTO `users` (`id`, `firstname`, `lastname`, `username`, `password`, `avatar`, `last_login`, `type`, `date_added`, `date_updated`) VALUES
(1, 'Adminstrator', 'Admin', 'admin', '0192023a7bbd73250516f069df18b500', 'uploads/1624240500_avatar.png', NULL, 1, '2021-01-20 14:02:37', '2021-06-21 09:55:07');


ALTER TABLE `jar_types`
  ADD PRIMARY KEY (`id`);


ALTER TABLE `sales`
  ADD PRIMARY KEY (`id`);


ALTER TABLE `sales_items`
  ADD PRIMARY KEY (`id`),
  ADD KEY `sales_id` (`sales_id`);


ALTER TABLE `system_info`
```

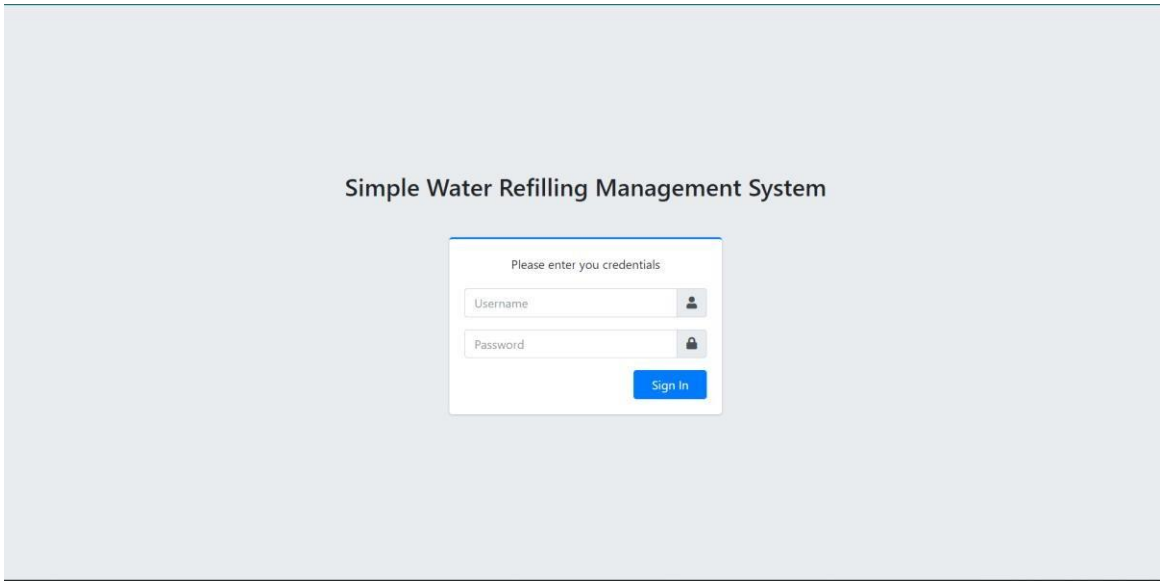
Screenshot 3: The Screenshot contains Description of table “users” the attributes of table users are id, firstname, lastname, username, password, avatar, last_login, type, date_added and date_updated.

```
ALTER TABLE `system_info`
  ADD PRIMARY KEY (`id`);


ALTER TABLE `users`
  ADD PRIMARY KEY (`id`);




ALTER TABLE `jar_types`
  MODIFY `id` int(30) NOT NULL AUTO_INCREMENT, AUTO_INCREMENT=5;


ALTER TABLE `sales`
  MODIFY `id` int(30) NOT NULL AUTO_INCREMENT, AUTO_INCREMENT=4;


ALTER TABLE `sales_items`
  MODIFY `id` int(30) NOT NULL AUTO_INCREMENT, AUTO_INCREMENT=9;


ALTER TABLE `system_info`
  MODIFY `id` int(30) NOT NULL AUTO_INCREMENT, AUTO_INCREMENT=15;


ALTER TABLE `users`
  MODIFY `id` int(50) NOT NULL AUTO_INCREMENT, AUTO_INCREMENT=6;


ALTER TABLE `sales_items`
  ADD CONSTRAINT `sales_items_ibfk_1` FOREIGN KEY (`sales_id`) REFERENCES `sales` (`id`) ON DELETE CASCADE;
COMMIT;
```

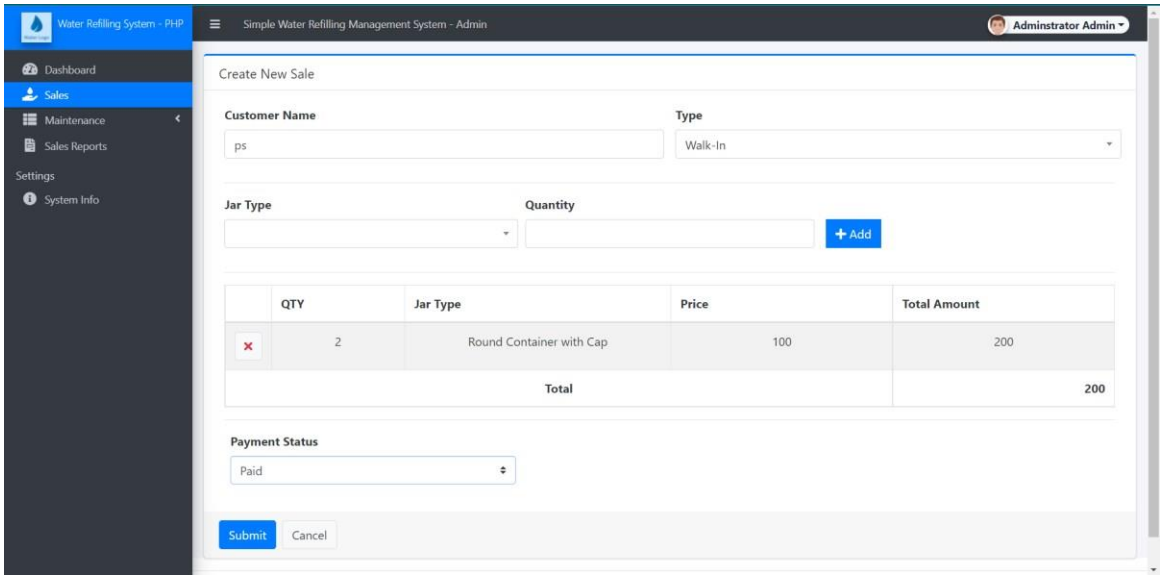
Screenshot 4: The Screenshot contains updation of tables jar_types, system_info, users, sales_items and sales. Primary and Foreign keys are defined in the above table updation.



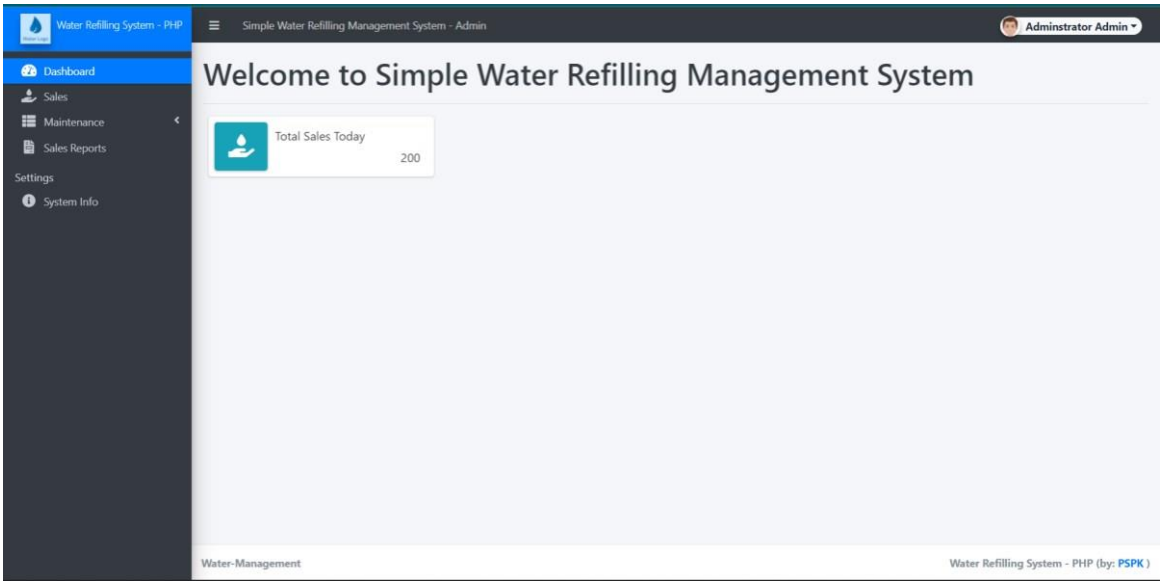
Screenshot 5: Showing XAMPP Control Panel. Which manages our localhost framework. Using this console, we are able to run the Website on our system with URL as our local-host address.



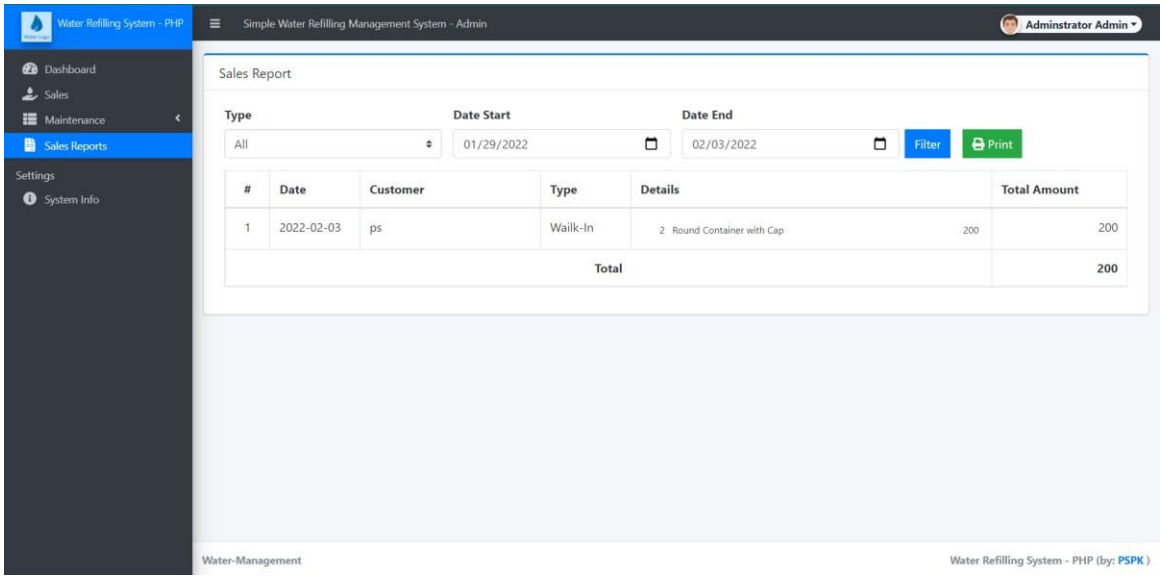
Screenshot 6: Showing login page. From here we can Login to the Application. By entering the correct Username & Password.



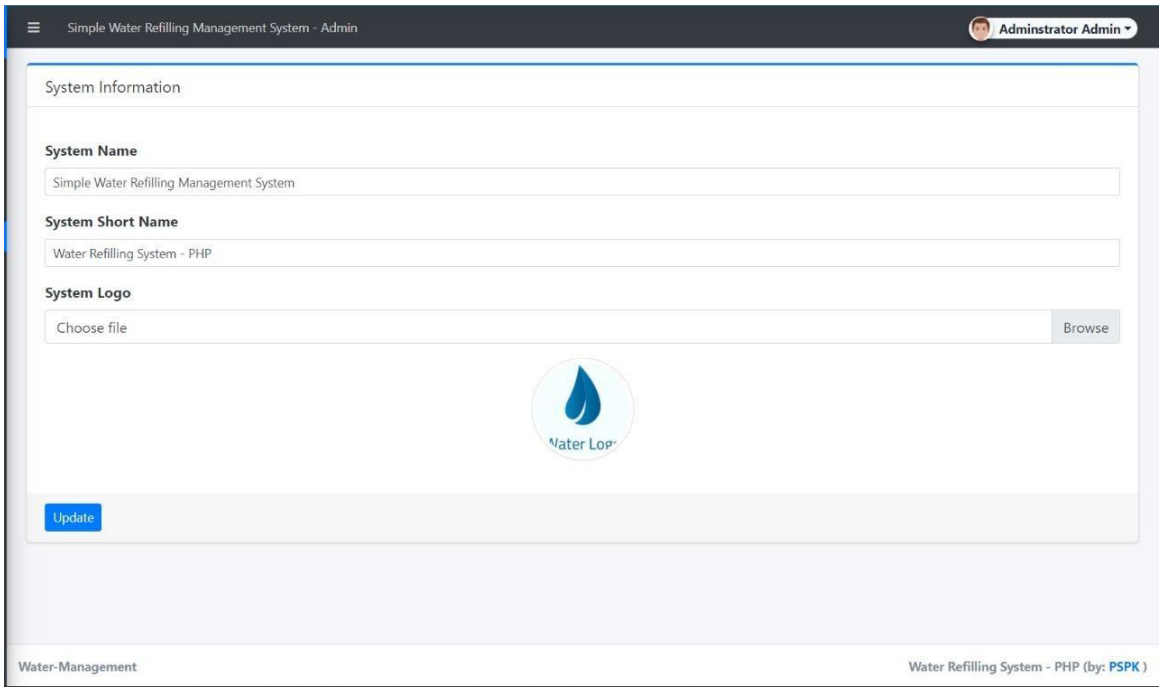
Screenshot 7: Showing create new sale section. Sales are easier to create from here. Moreover provides the flexibility to handle complete sales part from this single window present here.



Screenshot 8: Showing Total sales today. Helps to keep Track of the timely payment and dues so that timely payment and dues can be calculated and hence the financial management be done on the priority basis.



Screenshot 9: Showing Sales report. To know about the total Monthly income. Even queries can be set upon date_start, date_end and even at type of the jar. So that a track of income can be made available at one place.



Screenshot 10: Here we can update the details of the project from this window. We can update the company Logo and Name from here so that a complete scalability can be maintained.

CHAPTER 6

CONCLUSION AND FUTURE ENHANCEMENTS

To conclude the description about the project: The project is developed using PHP, MySQL, HTML, XAMPP and CSS on the requirement specification of the user and the analysis of the existing system, with flexibility for future enhancement.

The expanded functionality of today's software requires an appropriate approach towards software development. The application is developed and designed for people who want to manage their Water Can Business and get things on one place in this case is a website.

In future enhancements, we plan to add other features like:

• Generating reports for the selected sales, which will be sharable to others. On messaging platforms like WhatsApp

• Adding Backup feature so that we can retrieve our data if the server goes down or even if we have to transport from one domain to another.

• In Future we can also add feature of Bill generation for person walking to shop for easy usage to customers.

• Further we plan to store the database online so we can access it easily. By online we mean to invest in webserver and Domain names and getting a .com URL for the Website.

REFERENCES

1. https://stackoverflow.com/
2. https://en.wikipedia.org/wiki/
3. https://www.youtube.com/watch?v=1SnPKhCdlsU