International Journal of Research Publication and Reviews

Journal homepage: www.ijrpr.com ISSN 2582-7421

RESCUE DISPATCH MANAGEMENT SYSTEM

THANUSH M[1], *Dr. K. GEETHA*[2]

[1] BACHELOR OF COMPUTER APPLICATIONS, Sri Krishna Adithya College of Arts and Science, Coimbatore-641 042

[2] M.Sc., M.Phil., SET., Ph.D., Associate Professor, Head of the Department Sri Krishna Adithya College of Arts and Science, Coimbatore-641 042

1.1 INTRODUCTION

The project is entitled as Rescue Dispatch Management System which has developed in PHP as front and MYSQL server as back end.

The simple project is a Rescue Dispatch Management System. It is a web-based application project developed in PHP and MYSQL Database. The project provides an automated platform for managing the availability of the rescue Responders team and recording the incident report efficiently and effectively. It was developed with a pleasant user interface using Bootstrap Framework and AdminLTE Template. It also consists of user-friendly features and functionalities.

The Rescue Dispatch Management System can be only accessed by the dispatchers or the rescue management. The system stores the list of types of incidents, types of Responders team, and incidents reported. The system users are incharge of assigning responders to the location of the reported incident. The dispatcher can only assign the available team. The system also generates printable reports of Daily Incident Report and Daily Incident Report per Type. The application is easy to manage and consists of a front-end feature for managing the system information such as the system name, logo, etc. The system can be accessed by 2 types of user roles which are the Administrator and Team Head. The administrator has the privilege to access and manage all the features and functionalities of the system while the Team Headhave only limited permission.

1.2 MODULES

- Home Page
- Incident Type Management
- Respondent Type Management
- Team Management
- Report
- Encrypt
- Decrypt

MODULE DESCRIPTION

HOME

- It is a welcome page for users.

INCIDENT TYPE MANAGEMENT

- In this section, admin can manage Incident type (add/delete/update).
- The Module hold the incident id, incident name, description and status

➢ **Types of incident**

- Earthquakes,
- Tsunamis,
- Wild fires, ▪ Accident.

RESPONDENT TYPE MANAGEMENT

- In this section, admin can manage Respondent Type (add/delete/update).
- The modules holds the report id, report date time, remarks and location.

TEAM MANAGEMENT

- In this section, admin can manage Team Management (add/delete/update).
- The module holds the team id, code, team leader, respondent type and team status.

REPORT

In this section, admin can view the team details and check the Respondent reports (monthwise/year wise) in a particular

1.4 ABOUT THE SOFTWARE

FRONT END

○ **PHP**

PHP is a powerful server-side scripting language for creating dynamic and interactive websites. PHP widely used; free and efficient alternative to competitors such as soft's ASP.PHP is perfectly suited for Web development and can be embedded directly into the HTML code. The PHP syntax is similar to pearl and C.

PHP is open source that it is readily available and absolutely free. Stability, flexibility and speed are chief qualities that attract to choose PHP.PHP have multiple extensions and is extremely scalable.

○ **SERVER SIDESCRIPTING**

This server-side scripting is the most traditional and main target field for PHP. Programmer needs three things to make this work. Programmer need to run the web server, with a connected PHP installation. Programmer can access the PHP program output with a web browser, viewing the PHP page through the server.

BACK END

➢ **MYSQL**

MySQL is an open-source relational database management system (RDBMS),is developed, distributed and supported by MySQL. MySQL is a popular choice of database for use in web applications MySQL can be scaled by deploying it on more powerful hardware, such as a multi-processor server with gigabytes of memory. MySQL is easy to use, yet extremely powerful, secure, and scalable. And because of its small size and speed, it is the ideal database solution for Web sites

SYSTEM STUDY

2.1 EXITING SYSTEM

The existing system is offline process. Even though computerized systems are used in some places to not web-based and are very insecure and improperly managed. So, the current Rescue Dispatch Management is very uneconomical and inflexible to meet user demands. These are to be automated and an application is required to relate all of them relatively and logically so that the current system can be replaced and accepted without major changes and problems.

2.1.1 DRAWBACKS

- Lack of privacy.
- Less security.
- Less User friendly.
- Low coordination between people to team member.
- If this techniques is gone in the wrong hands like header, terrorists, criminals this can huge number of data huge file size so someone can suspect about it.

2.2 PROPOSED SYSTEM

The proposed of this research is to build a dispatch system that will help to speed up the rate of dispatching team in order to carter for emergency in the society. The objectives of the system includes: To provide fastest dispatching team services to the victims of any emergency incidents to decrease the amount of paper work must be file. To help the company ensure that provide an adequate amount of dispatching team for each of the service. To enable management, track the status of each dispatching team and the dispatching team activity

2.2.1ADVANTAGES

- The system is user friendly.
- Cost effective.
- Back up support-for data analysis.
- Secured of the hidden communication is high.
- Robustness increased manifold.
- The required information can be secured easily.
- Time will not Wasted in the process.
- High speed.

3.SYSTEM DESIGN AND DEVELOPMENT

3.1 FILE DESIGN

System design is the process of planning a new system to complement or altogether replace the old system. The purpose of the design phase is the first step in moving from the problem domain to the solution domain. The design of the system is the critical aspect that affects the quality of the software. System design is also called top-level design. The design phase translates the logical aspects of the system into physical aspects of the system.

3.2 INPUT DESIGN

Input Design is one of the most expensive phases of the operation of computerized system and is often the major problem of a system. A large number of problems with a system can usually be tracked backs to fault input design and method. Needless to say, therefore, that the input data is the life blood of a system and have to be analyzed and designed with utmost case and consideration. The decisions made during the input designer.The data, which is input to a computer – based information system, must be correct. If data is carelessly input and errors enter the system, it will lead to incorrect results whose consequences will be expensive and embarrassing to the designer. In data processing, the data entry operator often makes errors. This can be controlled by input design by using menu, interactive dialogue, consistent format etc. In this system the users are provided with user friendly pages to give the input and if the user gives any wrong input validations are done and message boxes are provided in the necessary places. The message specified in the message box is specified in a polite and in an informative manner. System is interactive dialogue, which simplifies the data entry or access, instead of remembering what to enter. User can choose from a list of options and type it in the cursor position. This will reduce the number of corrections while entering the data.

3.3OUTPUT DESIGN

The output design must be in such a way the user must able to understand the given details. So each detail given in the output should have some meaning in displaying the data. The output design is displayed in the form of data view. Output Design generally refers to the results and information's that areMgenerated by the system for many end-users, output is the main reason for developing the system and the basis on which they evaluate the usefulness of the application. The objective of a system finds its shape in terms of the output. The analysis of the objective of a system leads to determination of outputs. External outputs are those whose destination will be outside the organization and which require special attention as they project the image of the organization. Internal outputs are those whose destination is within the organization. It is to be carefully designed as they are the user's main interface with the system.

3.4 DATABASE DESIGN

The database design involves creation of tables that are represented in physical database as stored files. They have their own existence.

Each table constitute of rows and columns where each row can be viewed as record that consists of related information and column can be viewed as field of data of same type. The table is also designed with some position can have a null value. The database design of project is designed in such a way values are kept without redundancy and with normalized format. Refer the appendix for screen shots of database design.

3.5 CODE DESGIN

Designing code for a website project involves various aspects, including front-end development (HTML, CSS, JavaScript) and back-end development (server-side scripting, databases). Below are detailed considerations for code design in a website project:

3.5.1 FRONT-END (CLIENT-SIDE) CODE DESIGN:

- **HTML STRUCTURE:**

Maintain a clean and semantic HTML structure.
Use appropriate HTML5 elements for better accessibility and SEO.

- **CSS STYLES:**

Follow a modular and organized approach in writing CSS.
Consider using pre-processors (e.g., Sass, Less) for better maintainability.
Use responsive design principles for a mobile-friendly layout.

- **JAVASCRIPT:**

Implement modular JavaScript using functions and objects.
Utilize modern JavaScript features and frameworks (e.g., React, Vue, Angular) if applicable.
Optimize JavaScript code for performance and consider lazy loading for non-essential scripts.

- **USER EXPERIENCE (UX):**

Focus on providing a seamless and intuitive user experience.
Implement client-side validation for forms to enhance user interaction.
Consider using AJAX for asynchronous data loading to improve page responsiveness.

- **BROWSER COMPATIBILITY:**

Test your website on various browsers to ensure cross-browser compatibility. Include CSS vendor prefixes or use tools like Autoprefixer.

- **ACCESSIBILITY:**

Follow accessibility best practices to ensure your website is usable by people with disabilities. Use ARIA (Accessible Rich Internet Applications) attributes when necessary.

- **PERFORMANCE OPTIMIZATION:**

Optimize images and other assets to reduce page load times. Minify and concatenate CSS and JavaScript files. Leverage browser caching and compression techniques.

3.5.2 BACK-END (SERVER-SIDE) CODE DESIGN:

- **SERVER-SIDE SCRIPTING LANGUAGE (E.G., PHP):**

Adopt a modular approach in organizing PHP code. Use functions and classes for reusable components.Implement error handling and logging for debugging purposes.

- **DATABASE DESIGN:**

Design an efficient database schema with normalized tables. Use appropriate data types and constraints.Consider indexing for improved query performance.

- **SECURITY:**

Sanitize user inputs to prevent SQL injection and cross-site scripting (XSS). Implement secure session management and authentication.
Regularly update and patch server software.

- **APIS AND MICROSERVICES:**

If applicable, design a Restful API for interaction with external services or for front-end data retrieval. Consider implementing microservices for scalability and maintainability.

- **MIDDLEWARE AND FRAMEWORKS:**

Utilize server-side frameworks (e.g., Laravel, Django, Express) for Structured development. Implement middleware for handling common tasks like authentication, logging, and CORS.

- **CACHING AND PERFORMANCE:**

Use caching mechanisms (e.g., Redis, Memcached) for frequently accessed data. Optimize database queries and utilize query caching.

- **SCALABILITY:**

Design the back-end to be scalable, considering potential traffic growth.
Implement load balancing and horizontal scaling if needed.

- **LOGGING AND MONITORING:**

Implement logging to track errors and application events. Set up monitoring tools to track server performance, uptime,Version Control and
Collaboration.

- **VERSION CONTROL SYSTEM:**

Use a version control system (e.g., Git) to track code changes.
Follow branching strategies (e.g., GitFlow) for organized development.

- **DOCUMENTATION:**

Maintain comprehensive documentation for both front-end and back-end code. Include README files to guide developers on setting up and running the project.

- **COLLABORATION TOOLS:**

Use collaboration tools (e.g., project management platforms, communication tools) for effective team coordination.By incorporating these considerations into your website project’s code design, you can ensure a well-structured, secure, and scalable web application that provides an optimal user experience.

SYSTEM TESTING AND IMPLEMENTATION

4.1 SYSTEM TESTING

System testing for a rescue dispatch management system project involves a comprehensive evaluation of the system's functionality, performance, and reliability in a simulated real-world environment. This process includes analyzing requirements, developing a test plan, conducting functional testing to verify system functions such as dispatching resources and managing incidents, assessing user interface usability, testing integration of components, evaluating performance under various conditions, ensuring security measures, checking compatibility, regression testing, involving end-users in acceptance testing, reviewing documentation, and if applicable, testing localization capabilities. Through these steps, the system can be thoroughly tested to ensure it meets the needs of emergency responders and effectively aids in managing rescue operations.

4.2 TESTING METHODOLOGY

System testing is actually a series of different tests whose primary purpose is to fully exercise the computer-based system. System testing is the state of implementation that is aimed at assuring that the system works accuratssely and efficiently. Testing is the vital to the success of the system. System testing makes the logical assumption that if all the parts of the system are correct, the goal will be successfully achieved. The objective of testing is as follows:

- Testing is the process of executing a program with the intention of finding an error.
- A successful test is that one of the cover of undiscovered error.

❖ UNIT TESTING

Unit resting focuses on the verification efforts on the smallest unit of software design, the module. This is also known as “Module Testing”. The modules are tested separately. This testing is carried out during programming stage itself. In this testing each module is found to be working satisfactory as regard to the expected output from the module.

❖ VALIDATION TESTING

Here in the validation testing we want to check whether the given conditions to the text box are working correctly. Because in the name place we want to enter the characters and the special symbols only we should not enter the numbers in the name field. Here while on runtime we entered numeric values in the string specified columns of product inwards. It raises error. In this phase each module has been tested by wrong inputs, for example Employee Name should be a character as well as their age should be in numbers.

❖ INTEGRATION TESTING

Integration testing is a systematic testing for constructing the programs structure, while at the same time conducting the tests to uncover errors associated with in the interface. The objective is to take unit tested modules and build a program structure. Modules are combines and tested as a whole. Here correction is difficult because the cast experience of the entire program complicate the isolation cases. In this phase testing is done by how the system would interact with users and its User Interface flexibility.

❖ WHITEBOX TESTING

White box testing, sometimes called glass-box testing is a test case design method that uses the control structure of the procedural design to derive test cases. Using white box testing methods, the software engineer can derive test cases.

- Exercise all logical decisions on their true and false sides.
- Execute all loops at their boundaries and within their operational bounds.
- Exercise internal data structures to ensure their validity.

❖ BLACK BOX TESTING

Black box testing, also called behavioral testing, focuses on the functional requirements of the software. That is, black box testing enables the software engineer to derive sets of input conditions that will fully exercise all functional requirements for a program. Black box testing is not an alternative to white box techniques. Rather it is a complementary approach that is likely to uncover a different class of errors than white box methods. Black box testing attempts to find errors in the following categories. After preparing the test data the system under study was tested using test data. While testing the system by using test data errors were found and corrected. A series of tests were performed for the proposed system before the system was ready for implementation.

4.3 SYSTEM IMPLEMENTATION

The main focus of the analysis phase of Software development is on “What needs to be done”. The objects discovered during the analysis can serve as the framework or Design. The class’s attributes, methods and association identified during analysis must be designed for implementation language. New classes must be introduced to store intermediate results during the program execution. • Login Process has proper check for authentic user to login.

- No login process shall happen until unless username and password are provided.
- Users after login shall be able to view the proposal available for their events only.
- No staff member unless authenticate shall be able to login.

CONCLUSION

The Rescue Dispatch Management System project in PHP MYSQL is a valuable tool for emergency response teams to manage their operations effectively. The system offers a user-friendly interface for dispatchers to receive and manage emergency calls, allocate resources, and track the status of ongoing incidents.

The system is built on the PHP programming language, using the MYSQL database management system for data storage and retrieval. The project implements various features such as automatic location tracking, real-time updates, user authentication, and incident reporting.

Overall, the Rescue Dispatch Management System project is an essential tool for emergency response teams to streamline their operations and respond to emergencies promptly. By utilizing this system, dispatchers can allocate resources effectively, track incident progress, and ensure timely resolution of emergencies.

In conclusion, the Rescue Dispatch Management System project in PHP MYSQL is a valuable project that can significantly improve emergency response management. The system's features and functionalities are designed to enhance efficiency, accuracy, and speed in handling emergency calls and incidents.

5.1 FUTURE SCOPE AND ENHANCEMENT

The Rescue Dispatch Management System project in PHP MYSQL is an excellent tool for managing emergency response operations. However, it is always room for improvement and future enhancements. These are some possible future enhancements for this project:

- **Mobile Application**: A mobile application that integrates with the Rescue Dispatch Management System could be developed. This mobile application could allow first responders to receive real-time updates on emergency incidents, communicate with dispatchers, and update the incident status.

- **Predictive Analytics**: The Rescue Dispatch Management System could be enhanced with predictive analytics capabilities. This could help dispatchers anticipate and allocate resources for incidents that are likely to occur based on historical data.

- **Integration with Public Safety Networks**: The Rescue Dispatch Management System could be integrated with public safety networks, such as police and fire department databases. This integration would provide dispatchers with access to more comprehensive information about the incidents they are handling.

- **Artificial Intelligence:** The Rescue Dispatch Management System could be enhanced with artificial intelligence capabilities. This could help dispatchers identify and prioritize emergencies based on the severity of the incident and allocate resources accordingly.

- **Multilingual Support**: The Rescue Dispatch Management System could be enhanced with multilingual support. This would help dispatchers who speak different languages

BIBLIOGRAPHY

REFERENCE BOOKS

1. Matt Doyle, Beginning PHP 5.3, Published by Wiley Publishing, Inc. 10475 Cross point Boulevard, Indianapolis, IN 46256.
2. Larry Ullman, PHP and MySQL for Dynamic Web Sites Fourth Edition, Peach pits Press, 1249 Eighth Street, Berkeley, CA 94710.
3. Luke Welling and Laura Thompson PHP and MySQL Web Development Second
4. Edition, Same Publishing, 201 West 103rd Street, Indianapolis, Indiana 46290.

WEBSITES:

1. http://www.php.net/manual/en/
2. www.w3schools.com
3. www.tutorialpoint.com
4. www.bootstrap.com
5. www.mysql.com
6. www.php.net.