Project Report for Event Planning & Management System

[GitHub Repository Link](http://www.github.com)

**1. Introduction**

The **Event Planning & Management System** is a web-based application designed to streamline the management of events, guests, and resources. This report provides an in-depth overview of the project, including its objectives, implementation methodology, challenges encountered during the development process, and future enhancements.

**2. Project Goals**

The primary goals of this project are:

* To provide an easy-to-use platform for users to create and manage events.
* To allow users to manage guest lists, track RSVPs, and allocate resources.
* To ensure the system is scalable, secure, and user-friendly.

**3. Project Scope**

The scope of the project includes:

* Building a web-based interface for users to register and log in to the system.
* Providing features for event management (create, edit, delete, view events).
* Integrating guest management features (add, edit, delete guests).
* Allowing resource management for events (e.g., venues, equipment).
* Implementing a simple and responsive UI.

**4. Methodology**

**4.1 Development Approach**

The development of the system was carried out with the help of finely iterative **Agile methodology** that facilitated the cyclic development process based on the feedbacks collected in each cycle. The project was divided into multiple sprints, focusing on the following phases:

* **Sprint 1**: Assuming design of the project structure and the early-stage architecture of the databases.
* **Sprint 2**: Creating all the needs for events (creating an event, managing a guest list).
* **Sprint 3**: Resource management and improving the user interface are two key amendments.
* **Sprint 4**: Testing, debugging, and improving system security.

**4.2 Tools and Technologies**

The following tools and technologies were used in the development of the project:

* **Front-end**: HTML5, CSS3, Bootstrap
* **Back-end**: PHP, MySQL
* **Database**: MySQL (phpMyAdmin for database management)
* **Development Environment**: XAMPP (Apache, MySQL, PHP)
* **Version Control**: Git

**5. System Architecture**

The system follows a **client-server architecture**:

* **Client-side**: The user communicates with the system indirectly through using a web browser interface while the system receives information through HTTP request and sends it to the server.
* **Server-side**: Requests are processed by the server and written in PHP, the data is stored in the database, specifically MySQL.
* **Database**: Most of the system data included in the system such as user, events, guests, and resources are stored MySQL database.

**6. Features and Functionality**

**6.1 Event Management**

* Users can add, read, modify and delete events.
* Information about events includes the name, the date, time when this event happens and a brief description of it.

**6.2 Guest Management**

* One can include, modify and exclude any guest who has been linked to any given occasion.
* RSVP statuses are distinguishable (Invited, Accepted, Declined).

**6.3 Resource Management**

* There is permission whereby users can enter resources such as the venues, the catering services as well as equipment.
* There the resources are grouped and availability status is recorded in the system.

**7. Challenges and Solutions**

**7.1 Challenge 1: Database Connection Issues**

* **Solution**: In the development, there was a problem with the connection to the database at the initial stages of its development. This was done after correcting the db.php file configuration and confirming that MySQL services were on in XAMPP.

**7.2 Challenge 2: User Authentication and Security**

* **Solution**: The chapter one under the user authentication and security subtopic of the system addresses the issue of security of the system from unauthorized access.
* Providing safe end-user authentication was paramount. Passwords are encrypted by using password\_hash(), while to prevent SQL injection prepared statements are used.

**7.3 Challenge 3: Handling Large Guest Lists**

* **Solution**: Addressing the issue of data volume, pagination was introduced to the guest list in order to increase system performance.

**8. Future Enhancements**

Future improvements for the Event Planning & Management System include:

* **Email Notifications**: Notify the guests instantaneously every time an event is created and changes are made.
* **Analytics Dashboard**: event management, offer users statistical and analytical data concerning attendance, RSVP, and used resources.
* **Mobile Optimization**: Optimise the system for touch to making it quite effective when used on an Android phone or tablet.

**9. Conclusion**

In conclusion, the **management and planning of the system** deliver the goals and objectives of the project by offering an easy process in managing the events, guest lists and resources. During its development, the system has faced some concerns, but at the moment, it is actually functional, can scale and be prepared for enhancements in the future. The project showcases great comprehensiveness of full stack web development both in back end and front end.

**10. References**

* XAMPP Installation Guide: <https://www.apachefriends.org/index.html>
* PHP Documentation: <https://www.php.net/manual/en/index.php>
* MySQL Documentation: <https://dev.mysql.com/doc/>
* Bootstrap Documentation: <https://getbootstrap.com/>