# Introduction

Events are organized in many various forms such as Ceremonies, Meetings, Parties, Festivals, etc. The nature of each event might differ from other but they all have one common task that if left without proper supervision can ruin the whole experience of a great event that is Event Planning and management. To plan any event one needs to confirm a few details before the event happens. These are the date and time of event, the venue, event's schedule, catering, guest invitations and their accommodation to make sure everyone in the event enjoys.

I have taken this project on Event Management System in context of fulfilment of MCA. I have tried to make the complicated task of event management as simple as possible. I have tried to design the project in such a way that it can be further extended to achieve more. Although I cannot claim that the project is a full and complete package as the tasks in the event management far exceed the imagination and can continue to increase as the research goes on. That is why I have selected the more prominent tasks that are present.

## Basic Tasks

This project on Event Management is to ease up the tasks such as:

* Event's Nature (Festival, Meeting, Party, etc.)
* Event's Date and Time
* Event's Venue
* Event's Schedule
* Guest List
* Guest Invitations and responses
* Checklists for smaller tasks
* Catering and Menu
* User ratings and reviews on Events

All the above mentioned tasks are very important for a successful event. If any of the task is left unattended the whole event will be a disaster. That is why the project comes in handy, as the project will be in the form of a web app and using proper User Id and authentications one can manage multiple event at one convenient portal from anywhere with a working internet connection as all the data will be stored online into the server's database.

# Project Insights

This part of proposal is about the technical information such as *project category, technologies being used, etc*.

## Project Category

This project falls under the category of **RDBMS** *or Relational Database Management System*. Meaning that most of the project deals with a server-side database on which the CRUD operations (Create, Read, Update and Delete) are performed via a Client.

## Project Technology

The project being in the category of RDBMS software requires a dedicated **server** to function as the server will continuously provide the users with the data they need. Not just provide but also accept the data given to it by the server and respond accordingly. Hence I will be using the open source *Apache* server which is meant to be used in conjunction with **PHP**.

A database management system in the backend of our server is mandatory for us to keep working without worrying about the hassles of storing the data in a reliable and secure location. Also a DBMS provides other facilities like *Transaction management, Data recovery, User Authentication, Data Security, etc*. I will be using **MySQL** as the database backend for my project because it goes really well with the scripting language that we are using, *PHP*.

**PHP** being a server-side scripting language allows us to create web pages using regular HTML and **Apache** server serves these web pages effortlessly. Using **HTML** in conjunction with **CSS** gives beautiful and eye pleasing results as web pages. Hence the combination of PHP + HTML + CSS is very well suited to create a beautiful, easy to use and dynamic website. Also there are many UI centric CSS frameworks that can help build such websites. I am planning to use **Twitter's *Bootstrap,*** a CSS framework to achieve the same.

## System Requirements

To complete all tasks mentioned here, the requirements will be as following.

#### Hardware Requirements

|  |  |
| --- | --- |
| Processor | Intel Core i3 Processor or Higher |
| Ram | 4GB or above |
| Hard disk | 120GB or More |
| Network Connection | 1 Mbps or Higher |

#### Software Requirements

|  |  |
| --- | --- |
| Operating System | Windows 7 or Higher |
| Language | PHP, HTML, CSS, Java |
| IDE for PHP (Server Side + Website) | Netbeans 8.1 |
| Server | WAMP *(Windows Apache, MySQL and PHP)*  PHP 5.3 or higher with compatible MySQL. |

# Gantt chart

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Task** | **Week-1** | **Week-2** | **Week-3** | **Week-4** | **Week-5** | **Week-6** | **Week-7** |
| Analysis |  |  |  |  |  |  |  |
| Design |  |  |  |  |  |  |  |
| Coding |  |  |  |  |  |  |  |
| Testing |  |  |  |  |  |  |  |

# ER Diagrams

**Entity – Relationship Diagram**: This depicts relationship between data objects. The attribute of each data objects noted in the entity- relationship diagram can be described using a data object description. Data flow diagram serves two purposes:

* To provide an indication of how data are transformed as they move through the system.
* To depict the functions that transformation the data flow.

**Data Objects**: A data object is a representation of almost any composite information that must be understood by the software. By composite information, we mean something that has a number of different properties or attributes. A data object encapsulates data only there is no reference within a data object to operations that act on the data.

**Attributes**: Attributes define the properties of a data object and take on one of three different characteristics. They can be used to:

* Name an instance of data object.
* Describe the instance.
* Make reference to another instance in other table.

**Relationships**: Data objects are connected to one another in a variety of different ways. We can define a set of object relationship pairs that define the relevant relationships.

## Cardinality and Modality

Cardinality: The data model must be capable of representing the number of occurrences of objects in a given relationship. The cardinality of an object relationship pair is

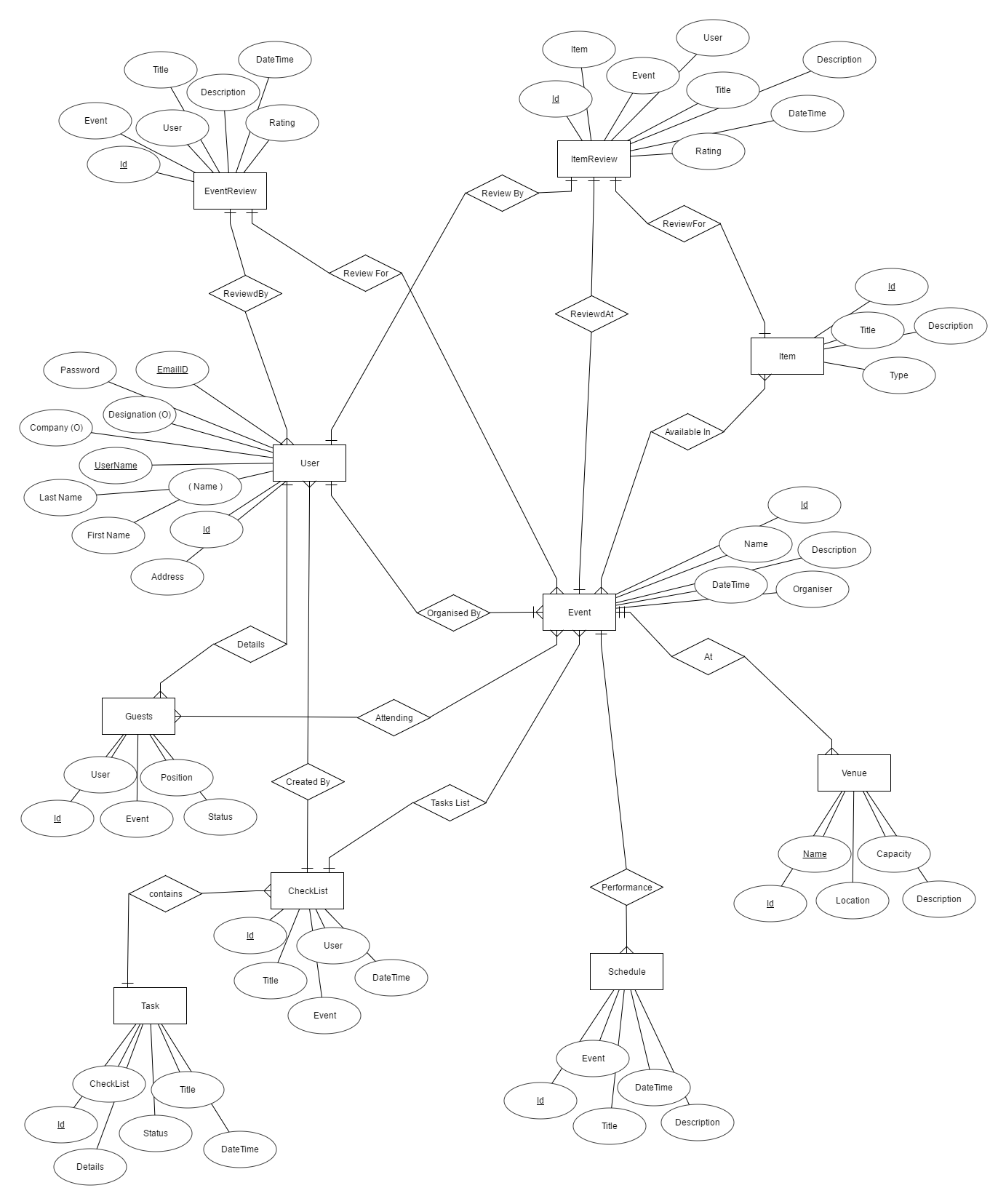
* One-T0-One (1:1): An occurrence of object ‘A’ can relate to one and only one occurrence of object ‘B’ and vice versa.
* One-To-Many (1:N): One occurrence of object ‘A’ can relate to one or may occurrences of object ‘B’ but an occurrence of object ‘B’ can relate to only one occurrence of object ‘A’.
* Many-To-Many (M: N): An occurrences of ‘B’ and an occurrence of ‘B’ can relate to one or many occurrence of ‘A’.

**Modality**: The modality of a relationship is zero if there is no explicit need for the relationship to occur or the relationship is optional. The Modality is one if the occurrence of the relationship is mandatory.

The object relationship pair can be represented graphically using the Entity Relationship Diagrams. A set of primary components are identified for the Entity Relationship Diagram,

1. Attributes
2. Relationships
3. Various Type Indicators.

The primary purpose of the Entity Relationship Diagram is to represent data objects and their relationships.



E-R Diagram Event Management System

# Database Tables

### Users

|  |  |
| --- | --- |
| Column Name | Data Type |
| ID | Integer |
| UserName | Varchar |
| Password | Varchar |
| EmailID | Varchar |
| FirstName | Varchar |
| LastName | Varchar |
| Company | Varchar |
| Designation | Varchar |
| Address | Varchar |

### Event

|  |  |
| --- | --- |
| Column Name | Data Type |
| ID | Integer |
| Name | Varchar |
| DateTime | DateTime |
| Description | Text |
| Organizer | Integer |

### Venue

|  |  |
| --- | --- |
| Column Name | Data Type |
| ID | Integer |
| Name | Varchar |
| Location | Varchar |
| Description | Varchar |
| Capacity | Integer |

### EventVenue

|  |  |
| --- | --- |
| Column Name | Data Type |
| ID | Integer |
| Event | Integer |
| Venue | Integer |

### Guests

|  |  |
| --- | --- |
| Column Name | Data Type |
| ID | Integer |
| User | Integer |
| Email | Varchar |
| Position | Varchar |
| Status | Varchar |

### Item

|  |  |
| --- | --- |
| Column Name | Data Type |
| ID | Integer |
| Type | Varchar |
| Title | Varchar |
| Description | Varchar |

### EventItems

|  |  |
| --- | --- |
| Column Name | Data Type |
| ID | Integer |
| Event | Integer |
| Item | Varchar |

### EventSchedule

|  |  |
| --- | --- |
| Column Name | Data Type |
| ID | Integer |
| Event | Integer |
| Title | Varchar |
| Description | Varchar |
| DateTime | DateTime |

### EventReview

|  |  |
| --- | --- |
| Column Name | Data Type |
| ID | Integer |
| User | Integer |
| Event | Varchar |
| Title | Varchar |
| Description | Text |
| Rating | Integer |
| DateTime | DateTime |

### ItemReview

|  |  |
| --- | --- |
| Column Name | Data Type |
| ID | Integer |
| User | Integer |
| Item | Integer |
| Title | Varchar |
| Description | Text |
| Rating | Integer |
| DateTime | DateTime |

### CheckList

|  |  |
| --- | --- |
| Column Name | Data Type |
| ID | Integer |
| User | Integer |
| Event | Integer |
| Title | Varchar |
| DateTime | DateTime |

### Task

|  |  |
| --- | --- |
| Column Name | Data Type |
| ID | Integer |
| CheckList | Integer |
| Title | Varchar |
| Details | Text |
| Satus | Varchar |
| DateTime | DateTime |

# Data Flow Diagrams

A DFD shows what kind of information will be input to and output from the system, where the data will come from and go to, and where the data will be stored. It does not show information about the timing of process or information about whether processes will operate in sequence or in parallel

## Context Level Data Flow Diagram



Context Level Event Management System

This depicts the working of proposed system from the user’s point of view. So any person without prior knowledge of programming can understand this as well.

## Data Flow while creating and managing an event



## Data Flow while Rating an Event or a specific Item

## 



# Modules

## Admin Module

There will be tasks in the project that require supervision. Such as to validate if the venue is accurately rated, or in fact available in the list. Same goes for user management tasks such as forgotten passwords, etc. So the admin module will cater to those needs and will have such tasks that only admins can perform. These admin account will be created by other admins or the super admin whose account is of the owner of the project.

But the major portion of the application will be from the perspective of the common user as an admin can also be required to perform such tasks.

## User Modules

Following are the modules that **every user** of the project can operate onto. These are to be created in a highly cohesive and loosely coupled manner to ensure further applications and enhancements in later stages.

### Event Creation

This will allow the users to create a new event with filling out a simple form that asks them to input the Event name, description, date and time.

### Venue Decision

Every event is bound to its unique criteria and that includes the nature of event, number of guests and the mood event needs to have. All these aspects are valuable inputs for us to decide on the location and venue for an event. A suitable venue is something that can accommodate all the guests and gives the feel of event. Like we cannot organize a formal business meeting or seminar in a garden as the garden will make the event more of a casual gathering not a formal one.

### Schedule

A good event manager always schedules all the small parts and performances etc for an event in a way that they do not interfere with each other and the guests are entertained till the end. The app will help user to keep track of such things in an elegant timeline view that can be modified beforehand.

### Guest List

Every event should have guest list to avoid problems like forgetting to invite someone to the event and later face the problems. This project helps a user keep track of all the guests and send invitations to them as well. On top of just sending the invite, the application also asks the guest to respond so the event manager can plan accordingly.

### Catering and Menu

The application will help an event manager to plan for the menu of the whole event and keep track of it as well.

### Custom Checklists

One can only predict a finite number of tasks in the vast concept of event management. There always are some unforeseen circumstances different for every event. To accommodate and handle them the application offers custom checklists.

### Decorations list

Visuals play a vital role in our regular life, but for an event the decorations are more than that. Decorations and arrangements are the most important part of all the events. So the application also helps us to keep track of such arrangements.

### Event Reviews

Any user who attended the particular event can later review the event so that the event planner can get insights on what can be done for future events of similar nature. Also by allowing users to review an event we are making the system more efficient.

### Item Reviews

Similar to the event reviews, a user can also rate and review particular items in an event. The items can be related to food or a small performance in the event.

### Event Report

After an event is completed, the members or in other words the organizers of said event always wish to know how the event went.

# Conclusion

This project is meant to be used as a complete package to manage and organize an event with ease because this will help the potential user to create an event, keep check list of random and small tasks, and schedule the event as well.

For this very project I have used Data Flow diagrams so the coding part of the SDLC will be easier.

By the completion of this project I will have a better understand of concepts like:

* Entity-Relationship diagram
* Using data flow diagrams
* Object Oriented Programing
* Client Server architecture
* Managing the data connection
* And much more…

# Future Scope

If we talk about a website now a days, people are accessing most of the websites on their mobile devices when they are traveling or working. So in our project which is to plan and manage events the user might not be near their desktops or laptops hence we should allow the user to access all the data in their own devices. For that I am planning an **Android Application** that can also be used to access the server. To develop a native android application I will need to use **Java** in *Android Studio,* the native IDE for android development.

Event management is a very vast area and there is always more and more modules to add in the project. The key modules that can be added in the project are:

* Money management
* Task assignment to others
* Contributing in organizing other’s event
* Creating a formal framework for reporting and sending them for future advisement.

The software as it is can be made live as a web application and android application on the google play store as well.

# Bibliography

To bring this project to verge of completion following books and websites are being referred.

* An integrated approach to Software Engineering by Pankaj Jalote
* System Analysis and Design by Elias M. Awad
* Introduction to SQL by Preeti Norton
* Sams Teach Yourself PHP, MySQL and Apache All in One by Julie Meloni
* <http://php.net/docs>
* <http://www.w3schools.com/php/>
* <https://www.tutorialspoint.com/php/>
* <https://stackoverflow.com>
* <http://www.w3schools.com/css/>
* <https://www.tutorialspoint.com/css/>