

**BCA Third Year**

**Semester : VI**

**Title of Project**

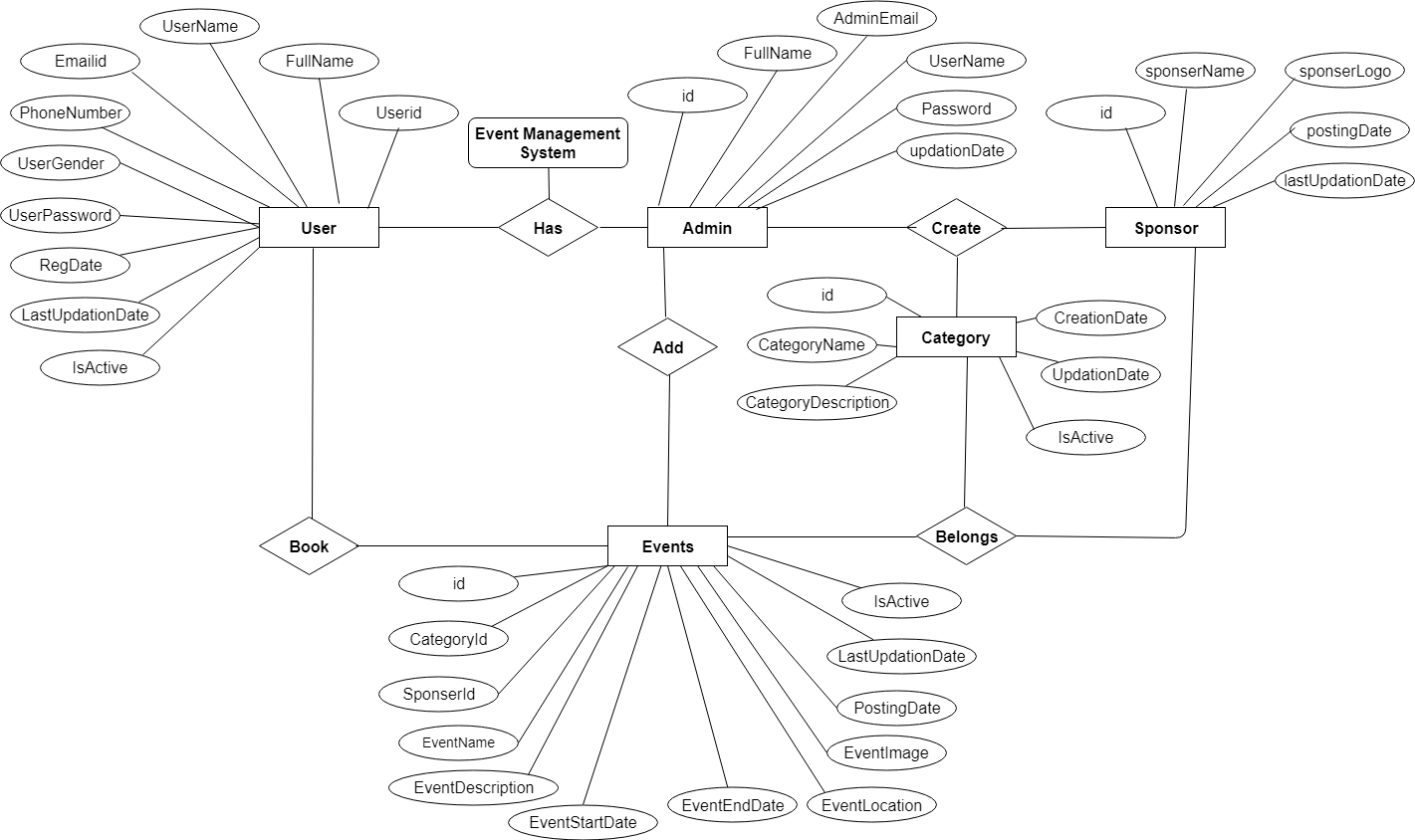
**Eventalist :- The Event Management Website**

Anurag Mahada (20210801070)  
Umesh Patil (20210801109)  
Vikram Jadhav (20210801133)

**Entity Relationship Diagram**

This is an ER model of a Eventalist -event management website. The entities are represented in rectangular boxes and relationships between different entities are represented by a diamond shaped box.

Here is the ER Diagram.



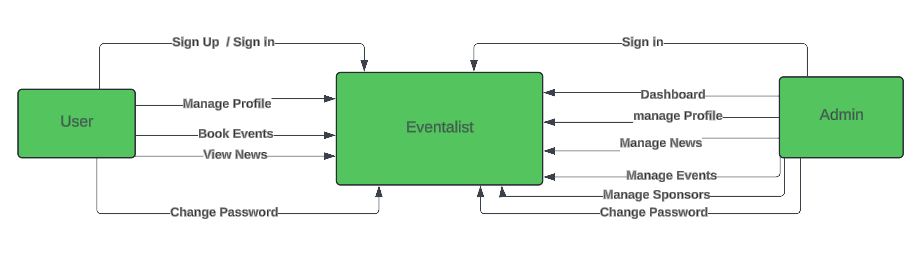
**Data flow Diagram (DFD)**

Following are application’s DFD level 0, 1, 2.

**Context level (0 Level) DFD :**

The Zero Level DFD is an abstract view of overall system. This is also called as Context Diagram in which entire system is represented as single process with

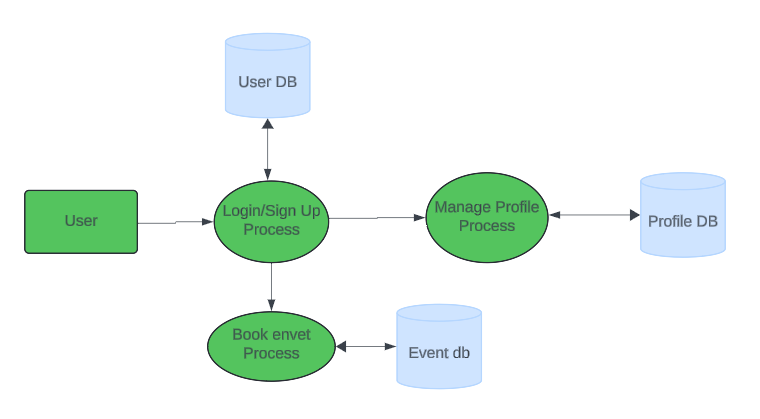
its relationship with external entities such as admin, user etc.



**Level 1 DFD :**

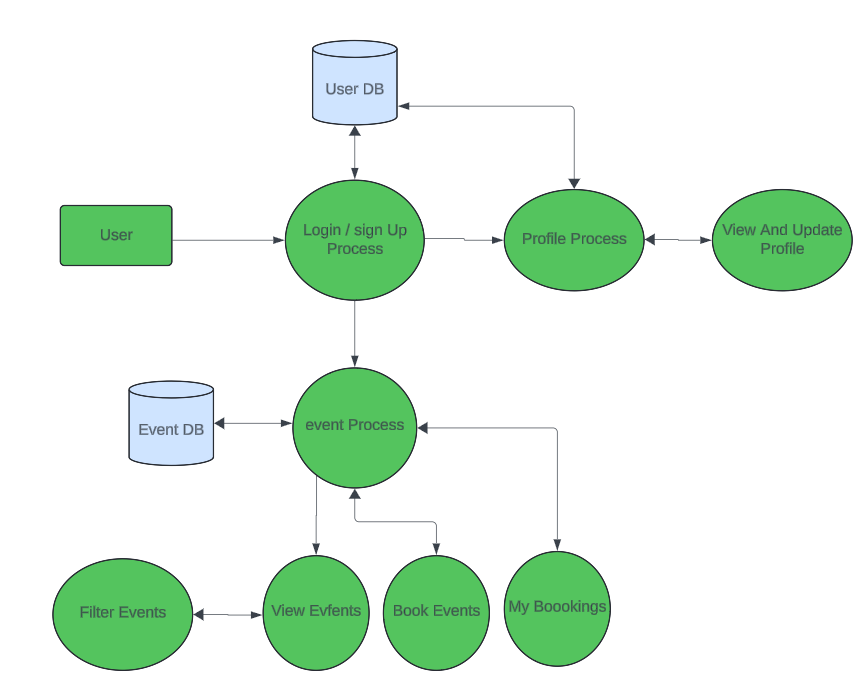
The first DFD of event management web shows more details of processing. Level 1 DFD list all the major sub processes that makes the entire system.

\



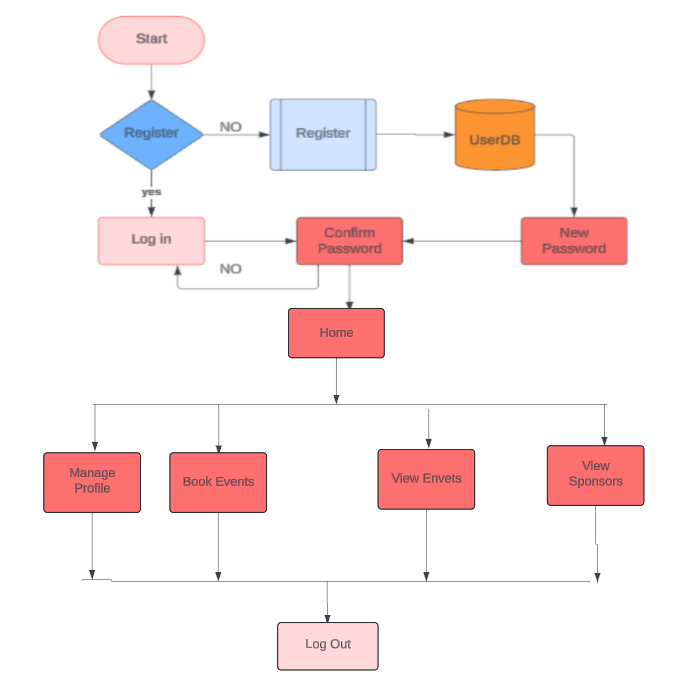
**Level 2 DFD :**

Level 2 DFD for event management web dives even deeper into the concept of Level 1 DFD.



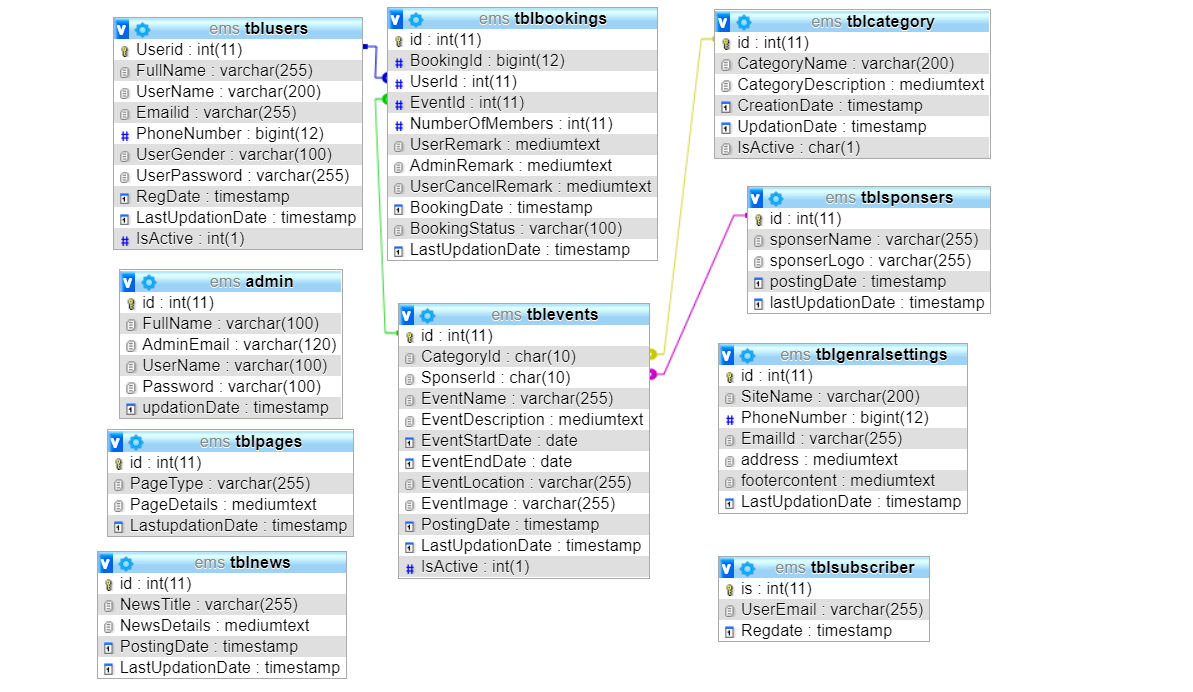
**Flowchart**

A flowchart is a visual representation of a process or algorithm. It is used to describe the sequence of steps and decisions required to perform a task.



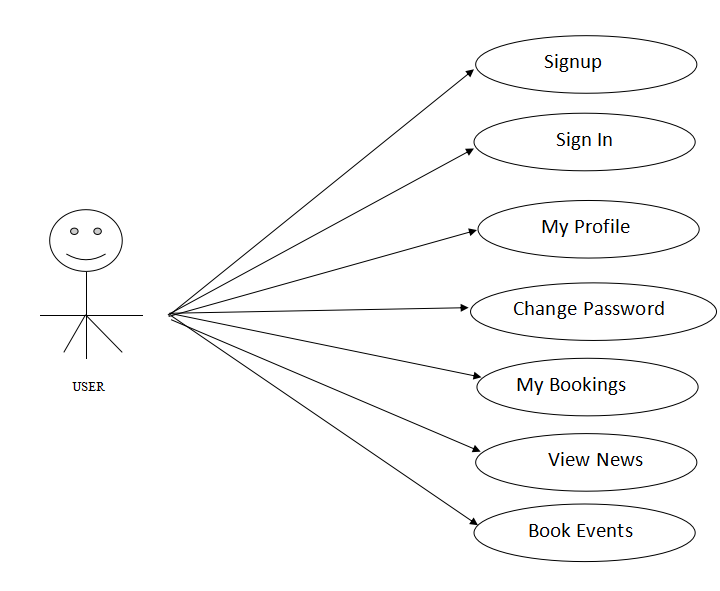
**Class Diagram**

a class diagram in the Unified Modeling Language (UML) is a type of static structure diagram that describes the structure of a system by showing the system's classes, their attributes, operations (or methods), and the relationships among objects.

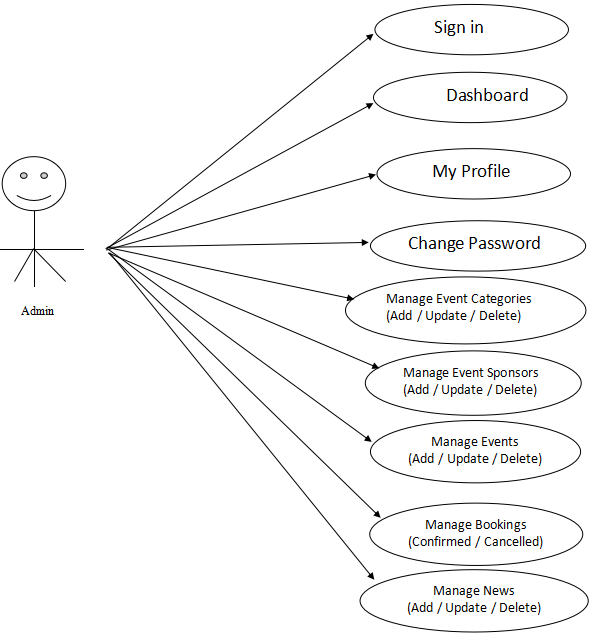


**Use Case Diagram**

A use case diagram is a type of Unified Modeling Language (UML) diagram that is used to model the interactions between a system and its actors.

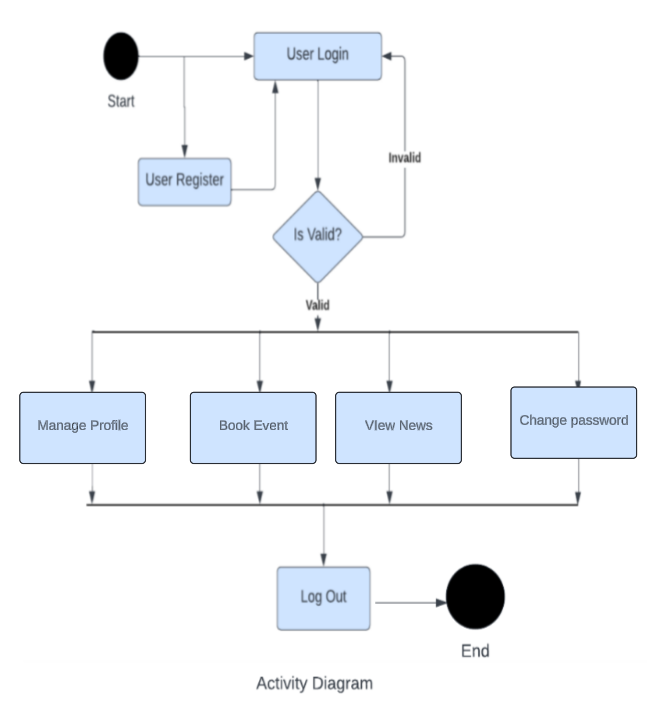


Use Case Diagram for User



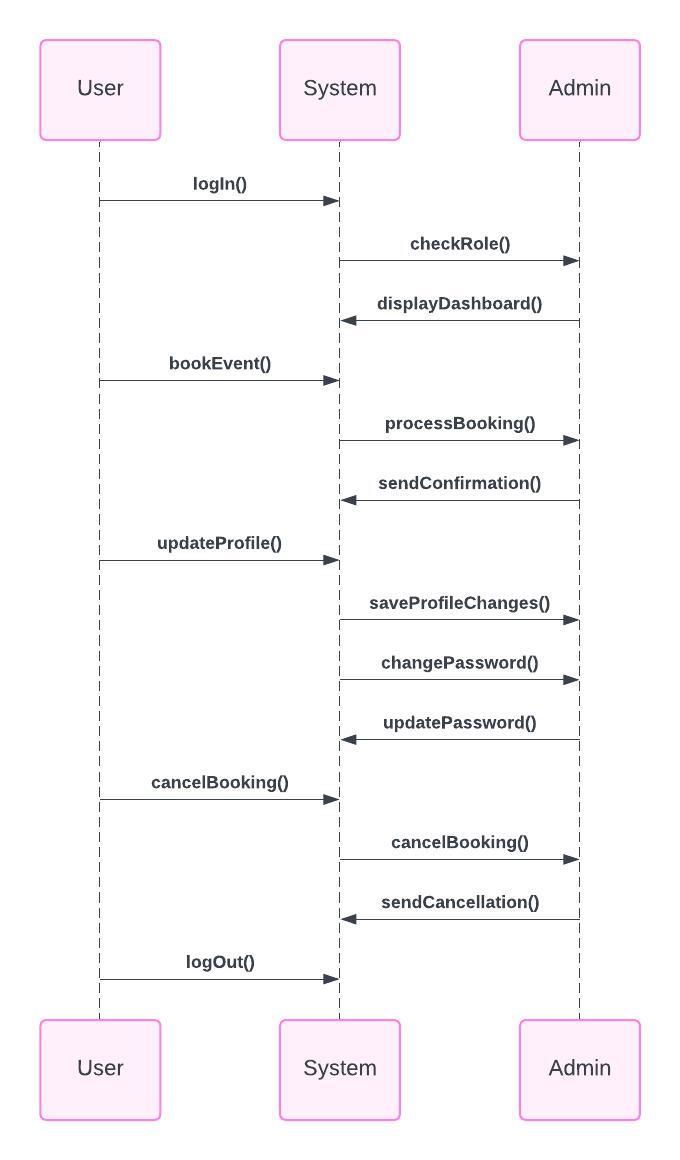
Use case Diagram For Admin

**Activity Diagram**

An activity diagram is a type of Unified Modeling Language (UML) diagram that is used to model the workflow of a system. It shows the flow of control between activities and the order in which they occur.

**Sequence Diagram**

A sequence diagram is a type of Unified Modeling Language (UML) diagram that is used to show the sequence of messages exchanged between objects and the order in which they occur.



**Pseudocode**

// Admin Module

class Admin {

function dashboard() {

// Fetch and display dashboard statistics

}

function manageCategory() {

// CRUD operations for event categories

}

function manageSponsors() {

// CRUD operations for sponsors

}

function manageEvents() {

// CRUD operations for events

}

function manageUsers() {

// View, update, and block registered users

}

function manageBooking() {

// Confirm or cancel booking

}

function manageNews() {

// CRUD operations for news

}

function websiteSettings() {

// Update about us and other general settings

}

function updateProfile() {

// Update admin profile

}

function changePassword() {

// Change admin password

}

function recoverPassword() {

// Admin password recovery

}

}

// User Module

class User {

function bookEvent() {

// Book an event

}

function updateProfile() {

// Update user profile

}

function changePassword() {

// Change user password

}

function cancelBooking() {

// Cancel a booking

}

function recoverPassword() {

// User password recovery

}

}

class GuestUser {

function viewGeneralInformation() {

// View general information

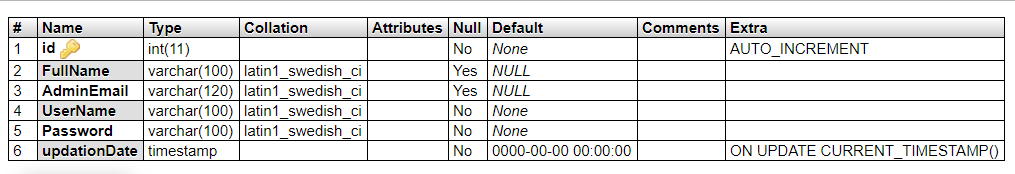
}

}

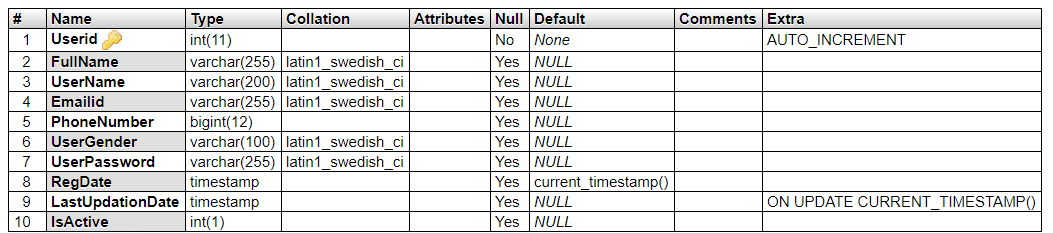
**DATA DICTIONARY**

The data in the system has to be stored and retrieved from database. Designing the database is part of system design. Data elements and data structures to be stored have been identified at analysis stage. They are structured and put together to design the data storage and retrieval system.

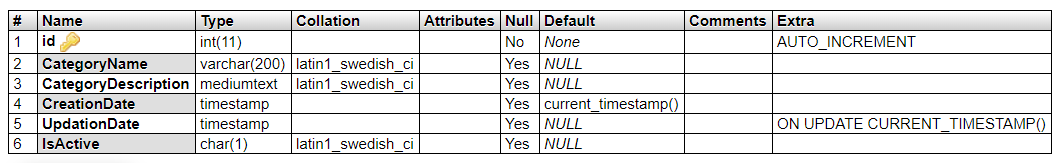
**tbladmin :** This table store the admin login details



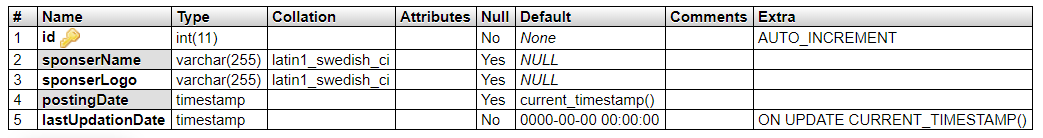
**tblusers:** This table store the user personal and login details.



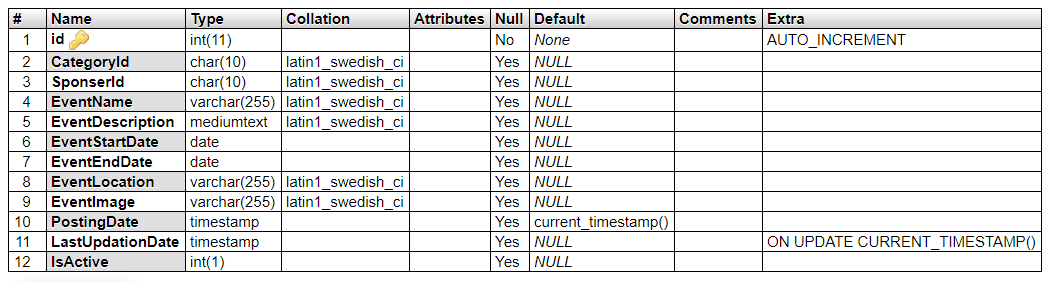
**tblcategory:** This table store the event category details.



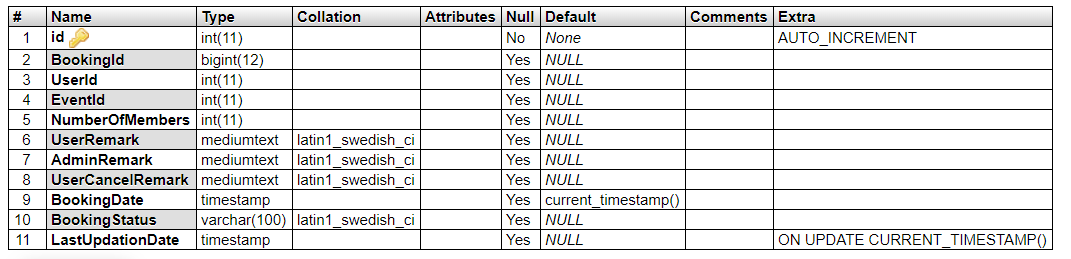
**tblsponsers:** This table store the event sponsor details.



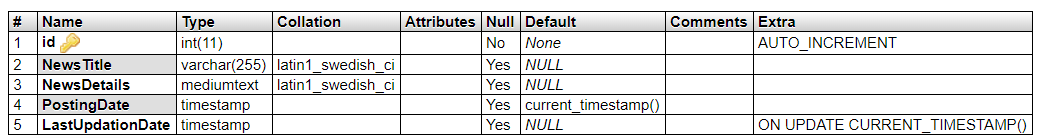
**tblevents:** This table store the event full details



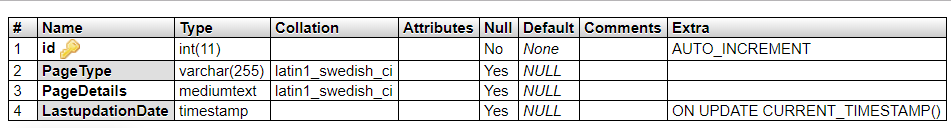
**tblbookings:** This table store the event booking details.



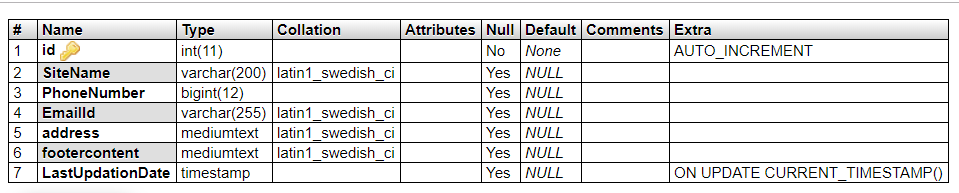
**tblnews:** This table store the news details



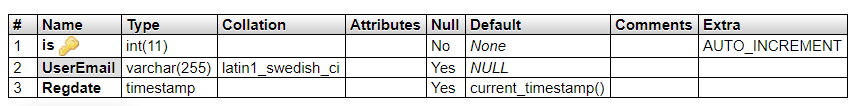
**tblnews:** This table store the pages info details.



**tblgenralsettings :** This table store the website general settings.



**tblsubscriber:** This table store the subscribers info.



**DECISION TABLE**

| **Action** | **Condition 1** | **Condition 2** | **Condition 3** | **Action 1** | **Action 2** | **Action 3** |
| --- | --- | --- | --- | --- | --- | --- |
| Admin | Logged In | Role: Admin | - | Dashboard | Manage Category | Manage Sponsors |
|  |  |  |  |  | Manage Events | Manage Users |
|  |  |  |  |  | Manage Booking | Manage News |
|  |  |  |  |  | Website Settings | Update Profile |
|  |  |  |  |  | Change Password | Recover Password |
| User | Logged In | Role: User | - | Book Event | Update Profile | Change Password |
|  |  |  |  |  | Cancel Booking | Recover Password |
| GuestUser | Any | Any | Any | View General Information | - | - |

Conditions:

Logged In: Whether the user is logged in.

Role: The role of the user (Admin/User).

Actions:

Admin: Admin functionalities.

User: User functionalities.

GuestUser: Functionalities available to GuestUse

**IMPLEMENTATION**

**INTRODUCTION:**

Implementation is the stage of the project when the theoretical design is turned out into a working system. Thus, it can be considered to be the most critical stage in achieving a successful new system and in giving the user, confidence that the new system will work and be effective. The implementation stage involves careful planning, investigation of the existing system and its constraints on implementation, designing of methods to achieve changeover and evaluation of changeover methods.

**TECHNOLOGIES USED**

**Programming Language**

**PHP**

* PHP stands for PHP: Hypertext Preprocessor
* PHP is a server-side scripting language, like ASP
* PHP scripts are executed on the server
* PHP supports many databases (MYSQL, Informix, Oracle, Sybase, Solid, Generic ODBC, etc.)
* PHP is an open source software
* PHP is free to download and use

**MYSQL**

* MYSQL is a database server
* MYSQL is ideal for both small and large applications
* MYSQL supports standard SQL
* MYSQL compiles on a number of platforms
* MYSQL is free to download and use

**CSS**

* Cascading Style Sheets (CSS)
* Simple mechanism