Title : Writing Problem Statement and SRS for the given project / system.

Objectives :

- Identify Project of enough complexity, which has at least 4-5 major functionalities.
- Write Problem Statement for System / Project
- Identify stakeholders, actors
- Prepare Software Requirement Specification Document.

Assignment 1A :

Problem Statement : Write a problem Statement for an Event Registration System

Theory :

What is a problem statement?

a. **Short and to the point:** A 1 – 3 page statement which everyone on an engineering project agrees with, saying, "This is what we are doing." By "everyone," we mean official project stakeholders, developers, and everyone else who should have some say-so in the project's outcome.

b. **A translation:** The problem statement *translates* the business case point of view of marketing people into the terminology and form used by the technical team who will develop a system, as a solution to a problem or need by paying customers. The problem statement will help those in business oriented non-technical jobs communicate effectively with the technical communities that support them, and help the technical communities understand, prioritize and delineate among needs, differentiators and wish-lists. It will also improve decision making speed and reduce rework resulting from a misunderstanding of the underlying reasons for what is requested by the product management team. These are all reasons why it's worthwhile to do this problem statement!

c. **Stays away from solving the problem:** The problem statement says, "*What* has to be done" for a development project to succeed – to meet the needs of its stakeholders who are external to the development. It does not say, "*How* that has to be done," unless those external stakeholders say so. This makes the problem statement become a tool for the development team to envision architectural and technology choices. Specifically, the problem statement matches an architectural document created by the architects for the project – that architecture document which follows will then say, point by point, how the proposed overall system design will meet the needs described in the problem

statement. Since both the problem statement and architectural document are fairly short, they have an “impedance match” which enables the architectural document to convince stakeholders of the viability of the proposed technical solution.

d. **Something done separately:** It is *not* a part of a larger requirements document, because it is a more conceptual document and is intended for a very wide audience. Also, the problem statement usually needs to be done first, so that the requirements reflect the concepts described in the problem statement. Thus, the problem statement does not fit in with the schedule of creation and organizational approvals for the related, thick requirements document.

e. **More than just the “short list” of requirements:** Furthermore, this is not just a consolidation of detailed requirements. The problem statement must say, for example, which really are the key requirements, and which ones are less important, something a requirements document may not say at all. And the problem statement explains why those are important to the success of the stakeholders.

f. **Serves a crucial role in software project success:** In other words, the problem statement is a format or tool for the stakeholders and developers to communicate in concise, plain language, about what tasks are being paid for, and what must be accomplished conceptually for the project to be a success. It is something everyone involved can tape on their office wall and know, “When we have done this, we made it!” Notice in the template for the problem statement, *below*, there is a place for stakeholders to sign-off on this document. This is very important! The problem statement forces all parties involved (including the development team’s leaders), to reach a conceptual agreement about what they are doing, and sign their names to this agreement.

Problem Statement for Event Registration System

We propose an **“Event Registration System”** that can manage all the events that are conducted in college/University. This system automatically generates certificates and issues it via mails. It can efficiently store, maintain, and retrieve data from its database. The system provides latest notifications to the user and intends on saving the users’ time by easing out the process of registering for events and staying in loop with the further process regarding these events. The application maintains data in a ***centralized way*** which is available to all the ***event managers***. It is very easy to manage ***historical data*** in databases. Participants can register for any happening event from anywhere. Event managers can keep records of participants and the

system can easily inform them by messages and emails. Thus the system shall simplify this task and make it more efficient. It intends on automating the process of registration.

Operation of the proposed system depends on its various users. These various user-types are mentioned below.

1. Admin
2. Manager
3. Normal User (Participants)/Volunteer

As no papers required so it reduces cost. It would be beneficial because only one time development efforts are required and no special skills or training required to run the system.
The proposed system is better than ordinary registration process and management because it is faster than older systems, the database keeps records and it is an onclick time saving system.

The system comprises of the following modules:
Registration : Participants can register online for any event.

Event : Event manager can generate events using this module. They can edit, delete the information of any generated event.

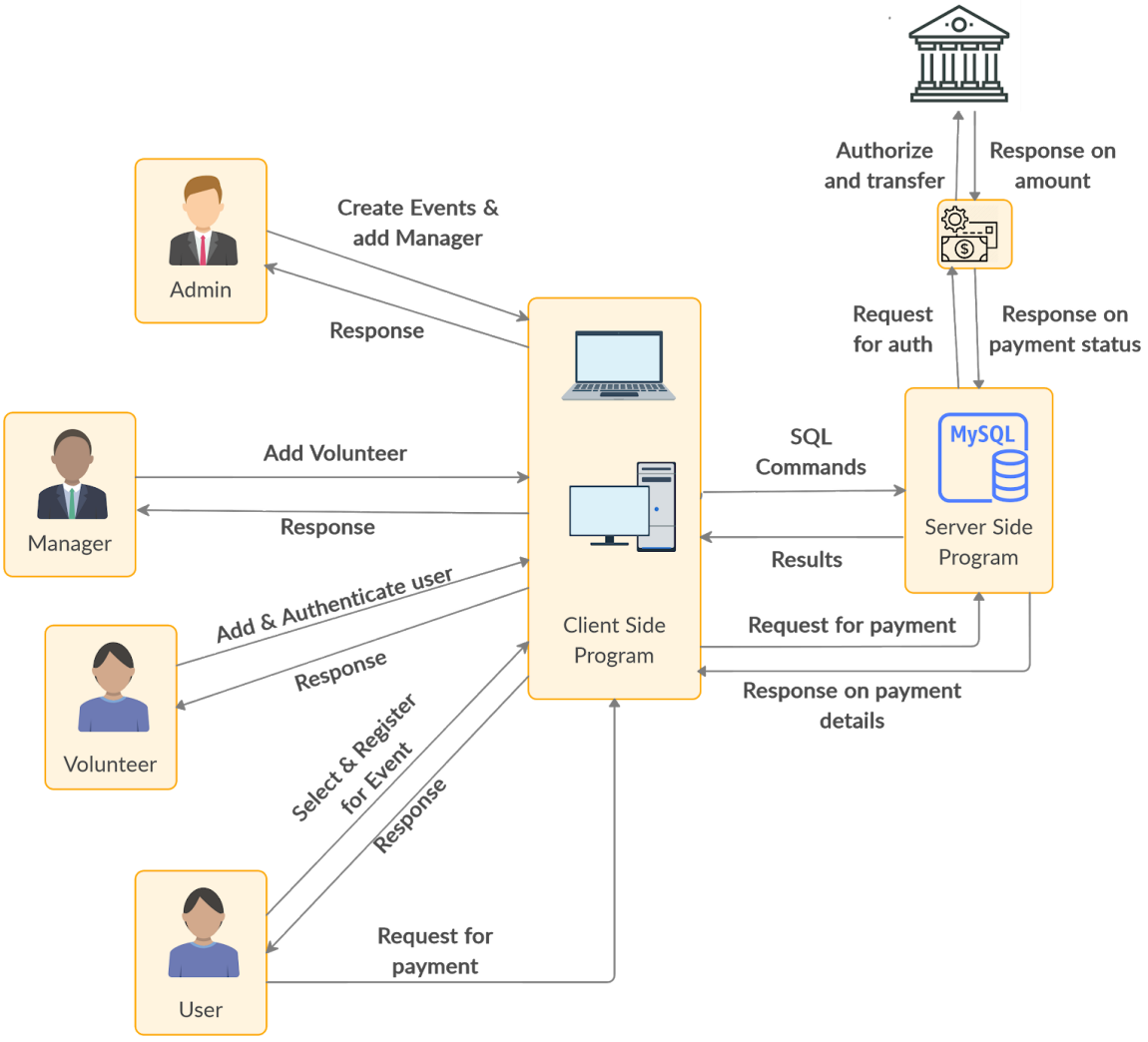
Certificate : System can generate certificates automatically and mail it to the particular participants.

Notification : Participants can get notifications about event timings, venue or any other updates.

User Module : Users can register for an event and can easily check event details, timings and can contact the event manager easily.

Volunteer Module : Volunteer can view the details of his/her event participants. Volunteers have to update the winner of his/her event in the system.

Admin Module : Database administrator has all the rights. He can add, edit, delete any information in the system. He can add new events, event managers, Volunteers as well as update them.



Use Case Scenarios

In the flow of the system, a typical use case scenario could be, Admin will make managers for different events and allocate Login ID. Thereafter Normal User shall enter his details i.e., Name, Enrollment Number, Course, Branch, College and select events. Then the manager will check user entries and allot registration id for authentication purpose. Managers can also change the date, time, venue of the events. Here, the manager is responsible for checking the team size and details and to restrict unauthorized people or wrong entries. After these all processes, Admin will check all details of the user, work of the manager and Admin can also make any changes at any stage.

Considering the use case of a user, the flow shall be as follows. The user shall input his username and password, which shall be checked as to whether he is registered in the database as a normal user. If the user is found in the database, he/she shall be directed to login. Once he/she enters the login details, i.e., username and password, the details shall be authenticated and checked. If found valid, he shall be directed to the profile to select events. Once the user selects events from the list of available events, he shall be redirected to the payment process. The payment shall display the event details and amount required to be paid, ask for confirmation to proceed and redirect the user to the payment gateway, if affirmative. In lieu of a successful payment, the user shall be brought to the logout page. In an alternate case, if the user is unable to login due to password memory, he can choose to reset his password using the Forgot Password link. If he/she clicks on the link, the user shall be requested to enter the email id to reset password. A reset password link shall be sent to the entered email id.
Otherwise, if the user is not found to be a registered user, he/she shall be directed to the new user registration. The registration process shall require the user to enter his/her details like username, password. After successful registration and authentication by the admin/volunteer/manager and updation of the database, the

user shall be directed to the login page, after which, the same process as mentioned earlier will be resumed.

We also consider the use case of adding a manager. A manager can be added by the admin. The admin shall send a request to add a manager to the system. He/she shall be required to enter the event required and the details of the manager, i.e., his id, username and password, and these details will be checked across the field constraints of the database. Upon successful verification, the manager with specified credentials shall be added to the database and the admin shall be sent a response of confirmation. The manager can then login with his allotted login id, username and password.

Following are the non-functional requirements of the system :

Performance Requirements:

The system needs to be reliable, generate appropriate error messages for transactions and load the web pages within seconds and also prevent system crash in the event of high traffic on the app/site.

Safety Requirements:

The Users should be authenticated based on their Google account or phone numbers. The database should be backed up well and properly at multiple sites to avoid loss of data in case of failures and prevent the loss of data due to clashing or concurrent transactions that can cause system failures.

Features:

The system needs to be integrated with the payment gateway to support in-app payment for events.

Assignment 1B :

Problem Statement : Writing an Software Requirement Specification(SRS) for an Event Registration Management System.

Theory :

A software requirements specification (SRS) is a document that captures complete description about how the system is expected to perform. It is usually signed off at the end of requirements engineering phase.

Qualities of SRS:

- Correct
- Unambiguous
- Complete
- Consistent
- Ranked for importance and/or stability
- Verifiable
- Modifiable
- Traceable

The SRS Generally contains the following.

1. **Introduction**
   - 1.1 Purpose
   - 1.2 Document Conventions
   - 1.3 Intended Audience and Reading Suggestions
   - 1.4 Product Scope
   - 1.5 References
2. **Overall Description**
   - 2.1 Product Perspective
   - 2.2 Product Functions
   - 2.3 User Classes and Characteristics
   - 2.4 Operating Environment
   - 2.5 Design and Implementation Constraints
   - 2.6 User Documentation
   - 2.7 Assumptions and Dependencies
3. **External Interface Requirements**
   - 3.1 User Interfaces
   - 3.2 Hardware Interfaces
   - 3.3 Software Interfaces
   - 3.4 Communications Interfaces
4. **System Features**

Software Requirements Specification

for

Event Registration System

Prepared by -

Rucha Shinde - Roll no - 43164

Shreya Thigale - Roll no - 43170

Riddhi Toshniwal - Roll no - 43171

Sakshi Tantak - Roll no - 43159

Pune Institute of Computer Technology

Table of Contents

Table of Contents

1. Introduction

1.1 Purpose

The Event Registration system is used to manage all the activity related to the event. In any event, many service providers work simultaneously and it is very hard to manage these providers. It is also important for the event organizer that he has all the contact details of these service providers so that he can contact them at any time to plan an event at a given time. To manage all these activities, we have developed this software. To get success in the event management business, users should have strong network contacts of service providers. These contacts are essentially providers of specific services who can be mobilized quickly to participate in any given event. To make an event successful, the event manager needs different service providers, like sound systems services, lighting providers, canteen services, stage construction and so on. In the present system, Event Company has to do all management work manually. They keep all payment information on papers. There is no system to check the past expenses on any event. To do this they have to check the payments register and this task is very time consuming and tiresome. Keeping this entire problem in mind, we have developed this system. This system helps the event management company to manage their paperwork online and they can also retrieve reports of the previous events they have completed.

1.2 Document Conventions

- Conventions for main title
  - Font face : Times New Roman
  - Font style : Bold
  - Font size : 18
- Convention for sub-title
  - Font face : Times New Roman
  - Font style : Bold
  - Font size : 14
- Convention for body
  - Font face : Times New Roman
  - Font size : 13

1.3 Product Scope

The scope of the project is clear to give a simple and attractive application to simplify the work as well as to reduce the efforts that are required in the process of the event registration system while doing it manually or we can say by doing it with old methods.

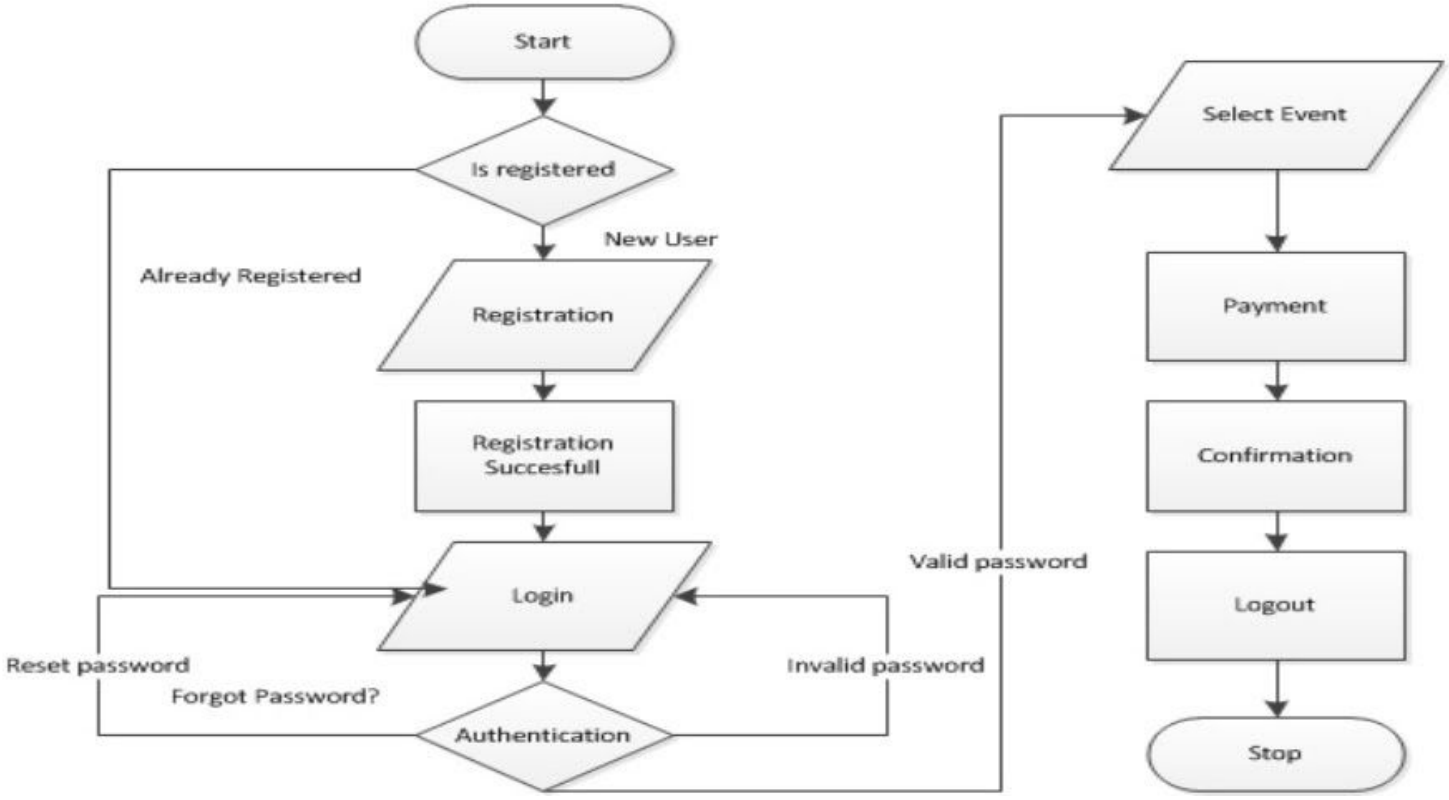
In this application we are able to save a database of all registrations made by the user. In this way we can keep track of records without human interference.
Registered users can do following activity

- Get verified via email.
- Register for the events.
- Login.
- Change Password.
- Update his/her own profile.
- Make required payments.
- Can see all the details in brief like listed categories, Upcoming Events, Old events, (with event details), his own registrations (with receipt/ticket).

2. Overall Description

2.1 Product Perspective

The software will be a new independent product, intended for the administration of the event management teams and other concerned users. The product will manage and import its data from MySQL Database backend and will be designed in JSP for the front-end. It will use JDBC for integration of the front-end and back-end database.
All the information and table will be selectively accessible by variou users involved in the interaction with the product, based on their privileges and rights over the data being accessed. The admin/developer will be able to access all of the database.
All the forms used in the product follow a clear and logical structure. Errors will be minimized via use of drop-down lists and radio/command buttons to eliminate excessive input of unstructured text. Management of data includes searching, adding, modifying and deleting.



2.2 Product Functions

The system comprises of the following modules:
Registration : Participants can register online for any event.

Event : Event manager can generate events using this module. They can edit, delete the information of any generated event.

Notification : Participants can get notifications about event timings, venue, certificate emails or any other updates regarding upcoming events, etc.

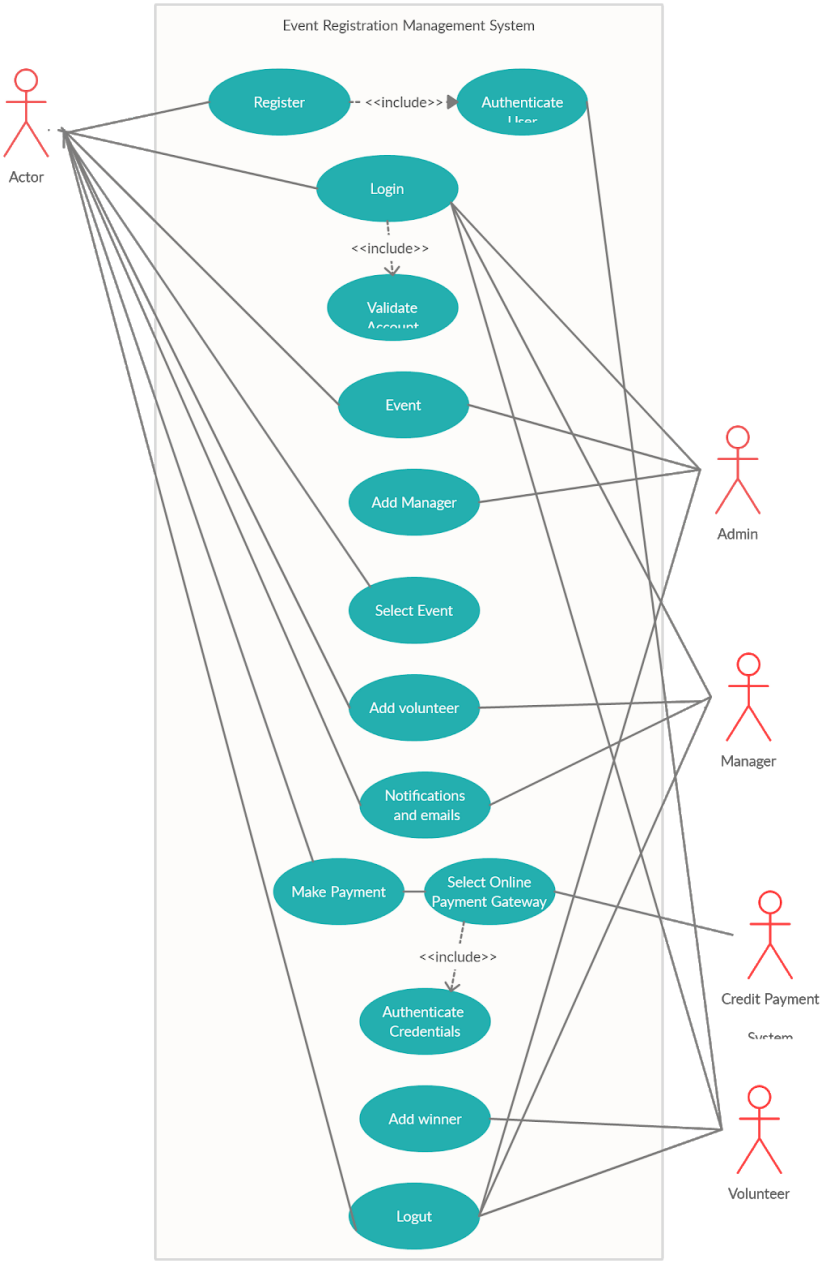
User Module : Users can register for an event and can verify himself/herself via email, check event details, timings, venue and contact the event manager easily.

Volunteer Module : Volunteer can view the details of participants of events under his/her jurisdiction. Volunteers have to update the winners (if any) of his/her event in the system.

Admin Module : Database administrator has all the rights. He can add, edit, delete any information in the system. He can add new events, event managers, volunteers as well as update them.

2.3 User Classes and Characteristics

An Event registration System is a system that serves the functionalities related to an event. This is proposed to be a web based system. Only authenticated users can enter into the system.



Based on the role of the individual using the system, he/she shall have the following services

- Admin
  - Database administrator.
  - Has all the privileges and rights over all the data.
  - Can add/delete/edit events, event managers, volunteers, users.
  - Can verify event managers, volunteers and users.
  - Can generate appropriate certificates for the event managers, volunteers and users via third party extensions.

- Event Manager
  - Can add/edit/remove events.
  - Can add/remove volunteers for the assigned event.
  - Can add/remove participants for his/her respective event.
  - Can verify volunteers and users.
  - Can add/edit/remove sub-events for the event using name, description, start-date, duration.
  - Can view details of the volunteers and users for his/her respective events.
  - Can check payment status of the users registered for the event.
  - Generate appropriate certificates for the volunteers and users via third party extensions.
  - Can change ownership of the events.

- Participants
  - Recommended to sign-up and get himself/herself verified on the site.
  - Sign-up will include name, roll number, email id, password.
  - Can change password and edit other details.
  - Can check his past events.
  - Can check upcoming events.
  - Can make payments, check their status and view their receipts.
  - Will receive notifications for running and upcoming events.

Following table shows the classes, their attributes and their respective functions.

Class	Attributes	Methods
Team	Id, event id, team size	select_event(), select_team(), payment()
User	Id, name, gender, contact no, enroll no, email id, college id, user type, username, password	login()
Admin	Id, uname, password	add_manager(), add_volunteer(), add_events(), check_participant(), check_team_details(), check_payment(), check_winner()
Manager	Id, uname, password	add_volunteer(), check_participant(), check_team_details(), check_payment(), check_winner()
Volunteer	Id, uname, password	check_participant(), check_team_details(), update_payment(), update_winner()
Normal User	Id, uname, password	select_event(), select_team(), select_payment()
Event	Id, event_name, organising_dept, manager_id, participation_amt, event_type, sub-events	
Winner	Id, event_id, user_id, position	

2.4 Operating Environment

Hardware specifications
To be able to run the system, the minimum requirements of the hardware for this system are:

- CPU 2.0 GHz or CPU (laptops) Core 2
- CPU (desktops) RAM 2 GB RAM
- HDD 60 GB min
- 7200 RPM6 GB or at least 10% free space (whichever is greater)
- The hardware used must have a competent firewall to secure the data in the system

Software Components / Applications:

- Language JSP.
- Database My SQL.
- Server XAMPP.

2.5 Design and Implementation Constraints

The system shall be bound by the following constraints :

- Only registered users will be able to book online event venues.
- Users will get any instant message through email address, not on mobile numbers.
- Every user will have his/her own private password of his/her account.
- To get important notifications via email, users must have to provide an email address and get it verified.
- For online payment, for example using cyber cash, customers will have to provide account numbers or other required details.
- Online meetings with the event organizer/manager are not available.
- Online photos will be available of event venues but to analyze the event venue, customers must have to physically visit the area where the event is going to be held.

2.6 Assumptions and Dependencies

- The user will be able to see available time slots for an event online.
- Most of the people will have internet connection to approach our web application, with a standard or widely used web-browser.
- Most of the people will visit our website to check out the interesting events.
- Online users will be able to get information like time slots, packages etc. of different venues.
- System will rely on the third-party payment gateways and services to carry out payments.

3. External Interface Requirements

3.1 User Interfaces

The application is very user friendly and uses a GUI interface implemented in JSP to communicate with the user. Various features are self – explanatory. Forms are easy to fill in and components can be added, removed and updated very easily. Drop downs and radio buttons are used to display all the components at once so that user can see all the components of a Particular type at once. One can just select the component and modify and remove the component.(based on the access control of the person).

There will be appropriate login, sign-up, dashboard and profile pages, payments processing page, and event board pages that will have a logical control flow and communicate with the user using navigation bars and seamless interfaces.

3.2 Hardware Interfaces

The application requires an error free system with good internet connectivity.All the information is maintained in a database.Based upon roles,restricted access to functionality of the system is allowed.

3.3 Software Interfaces

Windows 10 operating system is required to run the application.The database is managed and stored using MySQL Workbench.Also,Eclipse version 2018-19 is required.For the design of user interface HTML, CSS, Bootstrap 4, Javascript and JSP languages are needed.

3.4 Communications Interfaces

For communication purposes, the server used is Apache Tomcat 8.5. Google chrome or IE8 web browser can be used to access the application. The users can access the database using JDBC.

Overview of interfaces and requirements :

Operating System	Windows 10
Interface	JDBC
Database	MySQL
User Interface Design	HTML, CSS, Bootstrap 4, Javascript, JSP
Web Browser	Google Chrome, IE8
Server	Apache Tomcat 8.5
Software	Mysql Workbench, Eclipse version-2018-09

4. System Features

1- Registration for an event:
An event registration system allows online registration for an event in a dynamic and easy way. It helps organizations to streamline all the registrations and serve them properly. A proper maintained list makes it easy to monitor the registered people. Also the expectations of the registered people can be kept up to the mark by providing them best services for their attendees.

2- Payment processing of an event:
System provides the facility of fast and secure online payment processing of an event. It allows the event management to get payment directly in their account from anywhere in the world.

3- On-site functionality of an event:
It allows the on-site functionality of an event such as management of events, creates the list of attendees, updates the event , and allows to manage the complete planning of an event.

4- Increase the accessibility:
It is easily accessible to management as well as the attendee.It allows the web as well as mobile registration to the attendee and on the other side it also allows complete access to the management through web or mobile. Increased accessibility can improve the working of organization and profit can be more easily expanded.

5- Scalability of an event:
System allows changing the scalability of an event according to the need and requirement of an event as well as the organization. It is a helping tool to manage the volunteers for events.

5. Requirements

5.1. Performance Requirements

1. The performance of the system should be fast and accurate
2. Allow only authenticated users to connect in this system.
3. The user interface should provide appropriate error messages for invalid input.
4. Allow valid users to login and logout.
5. The system shall allow the user to sign in based upon email and password.
6. The user interface of the system shall be easy to use and shall make use of selectable fields wherever possible instead of fields that require the user to type in data.
7. Support mass customized template to reuse.

5.2 Safety Requirement

The database may get crashed at any certain time due to virus or operating system failure. Therefore, it is required to take the database backup so that the database is not lost. Proper UPS/inverter facility should be there in case of power supply failure.

5.3 Security Requirement

- Participants can just read information but they cannot edit or modify anything except their personal and some other information such as event details.
- System will have different types of users and every user has access constraints.
- Proper user authentication should be provided.
- No one should be able to hack users' passwords.
- There should be separate accounts for admin and members such that no member can access the database and only admin has the rights to update the database.

5.4 User Requirements

- Normal Requirements
    - Web-Based Interfaces , user friendly
    - The application can be accessed from any computer that has Internet access
    - Maintain a database of all information in this system

- Expected Requirements
    - Maintain a database of all information in the Event Registration System.
    - The system shall allow the user to sign in based upon email and password.
    - The user interface of the system shall be easy to use and shall make use of selectable fields wherever possible instead of fields that require the user to type in data.

- Exciting Requirements
    - Even unauthenticated users can view a list of events that are being organized.
    - Personalized user dashboard

5.1 Software Quality Attributes

The following are the software quality attributes of product:

- Convenience & Usability
- Easy update and retrieval
- Modifiability
- Performance
- Security
- Scalability

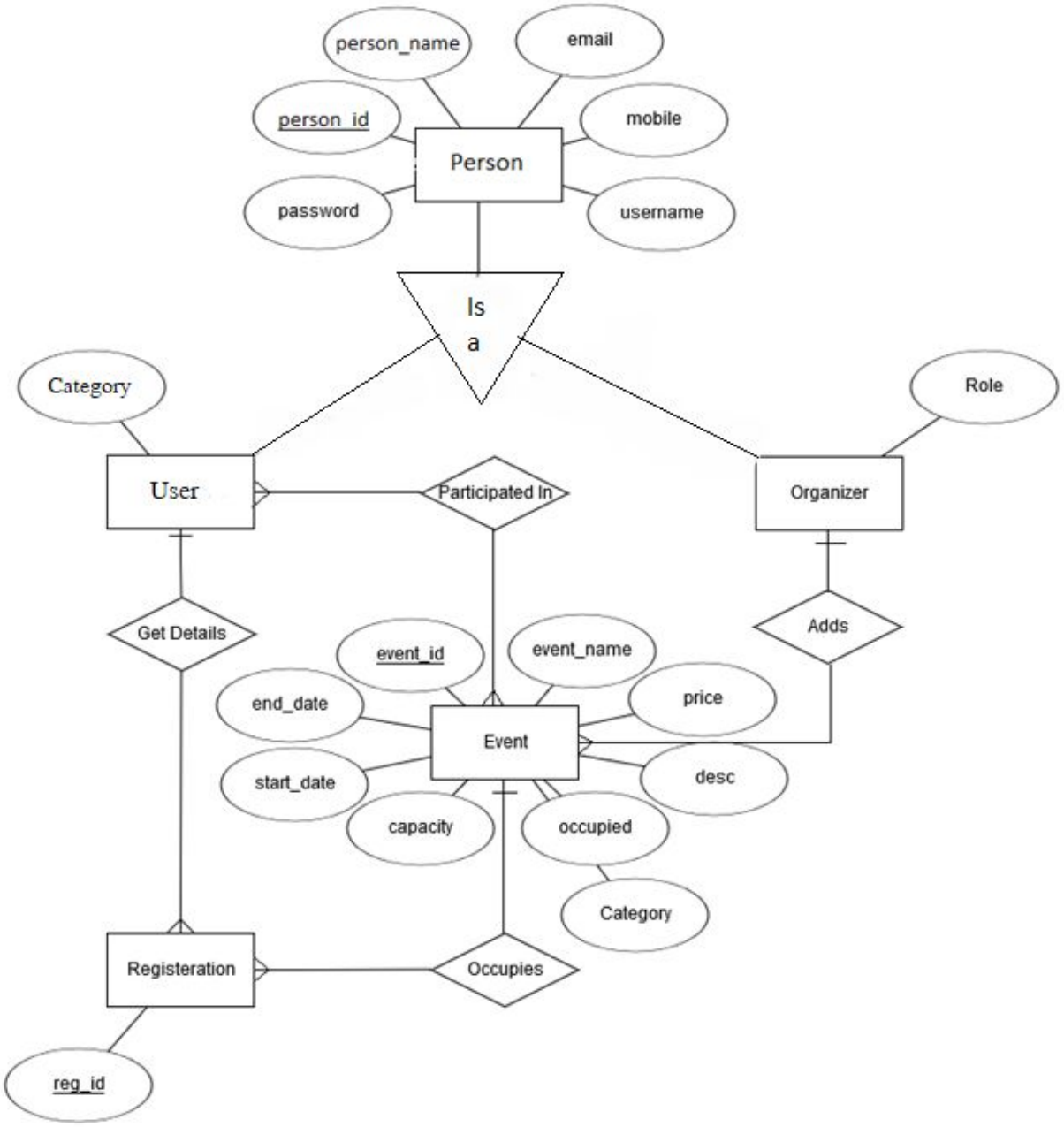
Appendix A: Glossary

Acronyms

Acronym	Definition
SRS	Software Requirement Specification
ERS	Event Registration System
QFD	Quality Function Deployment
ER	Entity Relationship

Appendix B: Analysis Models

Schema Diagram



Appendix C: References

References

- **SRS** - http://www.wikihow.com/Write-Software-Documentation.
- **Schema Diagram** - https://docs.oracle.com/cd/B28359_01/server.111/b28328/diagrams.htm
- https://headchannel.co.uk/top-5-features-of-an-event-management-software-321

Conclusion :

We have understood how to approach a project with an objective and structured approach. We understand how to begin this process with the development of sophisticated Problem Statement and SRS documents. We have thus implemented accepted standards and procedures to develop a Problem Statement and SRS for project idea we have picked (Event Registration System).