**YOGA CLUBS & COURSES MANAGEMENT SYSTEM**

**1.Introduction**

**1.1 Category**

Web Application

1.**2** **Purpose**

The YOGA CLUBS & COURSES is a web-based application intended for end users to register for offline and online yoga courses as well as to join online clubs for daily practice of yoga easily at their home. The main objective of this application is to make registration process easier and to build interactive communities of fit people. Here end users who are unaware of yoga techniques can browse the yoga courses based on what they want to learn. It means this application is helpful for end users to learn yoga taught in Indian culture which will be a healthy habit for them that can be followed at home easily using our web app. Admin will have all the rights to bear or change or suspend the rules and regulations.

**1.3 Scope**

It will integrate the benefits of a physical yoga center with convenience of a 'no-physical-bar' virtual yoga practices environment, minus the change hazards and expenses. This will provide more flexibility in the existing fitness platform structures. It provides a means of collaborative learning for people and also promotes safety being home.

**1.3.1 Features:**

1)Join yoga community in your area for daily practices.

2)Search and register for yoga courses(online/offline).

3)User profile with their health details.

4)Payments pages.

5)Feedback forms.

**1.4 Definitions:**

YCC --> Yoga Clubs & Courses

SRS --> Software Requirement Specification

GUI---> Graphical User Interface

Portal--> Personalized Website

Stack-holder--> The person who will participate in the System. And Owner of system

Ex. User, Administrator, Yoga Teacher

UML---> Software Engineering Notation for visualizing System in the form diagrams

SSL---> Secure Socket Layer used for providing restricted access to application.

BOD---> Board of Directors (Management).

RDBMS --> Relational Database Management System.

CLUSTERS---> Group of independent servers.

**1.5 Overview:**

Not only this system helps end users to practice daily yoga at their home with their companions but also, it provides an easy solution for users to find the online yoga courses without going anywhere with the best yoga teachers depending on their reviews. The term client/server refers primarily to an architecture or logical division of responsibilities, the client is the application (also known as the front-end), and the server is the RDBMS (also known as the back-end). A client/server system is a distributed system in which some sites are client sites and others are server sites. All the data resides at the server sites. All applications execute at the client sites.

**2 Functional Requirement:**

This section provides requirement overview of the system. Various functional modules that can be implemented by the system will be-

**2.1 Description:**

**2.1.1 Registration:**

if user wants to join the yoga community then he/she must be registered, Unregistered user cannot get access.

**2.1.2 Login:**

User logins to the system by using valid user id and password to find the yoga class.

**2.1.3 Browse courses:**

End User can Browse courses and yoga teachers, locations also.

**2.1.4 Updating courses history:**

Updating track record of completed course after successful completion of each course.

**2.1.5 Alerts:**

**2.1.6 Small videos:**

**3 Technical Issues:**

This system will work on client-Server architecture. It will require an internet server. The system should support some commonly used browser such as Chrome etc. Interface Requirement Various interfaces for the class could be

1.Login Page,

2.Registration form

There will be a screen displaying information about category of FAQs that the site having.

The users may select the different options which will be open in another screen as

1.Login Page

2.Registration Form

3.course details page (include locations)

4.Yoga teachers details page

5.course progress

6.Account Settings

7.Payment Gateways

**4 Hardware Interface:**

The System must run over the internet and the hardware required to connect to the internet will be the hardware interface for the system.

e.g., modem, WAN, LAN

Specialized Server Infrastructure Hardware the system should use distributed servers i.e., cloud for managing large amount of data so as to make it appear as single unit for end-user. The system should have proper clusters for backup.

**5 Software Interface:**

The system is on server so it requires the scripting language which is JSP. The system should be able to exchange data using XML, JASON. The system requires MySQL database also to store the transactions of the system. System also require DNS (Domain Name space) for the naming on the internet.

http://www.ycc.in

At the end-user need web browser to interact with the system.

**6 Performance Requirement:**

There is no performance requirement in this system, because the server request and response to client is totally based on internet connection of end-user.

**7 Design Constrains:**

This system should be developed using Standard Web Page Development Tool, which conforms GUI standards such like HTML, XML, JSON, etc.

The system should support various RDMS and Cloud Technologies.

**8 Non-Functional Requirements**

**8.1 Security:**

The System use SSL (Secure Socket Layer) in all transactions that include any confidential user information. The system must automatically log out all user after a period of inactivity. The system should not leave any cookies on the user's computer containing user’s password. The system's back-end servers shall only be accessible to authenticated administrators. Sensitive data will be encrypted before being sent over insecure connections like internet. The proper firewalls should be developed to avoid intrusions from the internal or external sources.

**8.2 Reliability:**

The system provides storage of all databases on redundant computers with automatic switchover. The main pillar of reliability of the system is the backup of the database which is continuously maintained and update to reflect the most recent changes.

**8.3 Availability:**

The system should be available at all times. Meaning the user can access it using web browser, only restricted by the down time of the server on which the system runs. In case of a of a hardware failure or database corruption, a replacement page will be shown.

24 \* 7 availability

**8.4 Maintainability:**

A commercial database is used for maintaining the database and application server takes care of the site. The maintainability can be done efficiently.

**8.5 Portability:**

The application is HTML and scripting language based (JavaScript). So, the end user part is fully portable and any system using any web browser should be able to use the features of the system, including any hardware platform that is available or will be available in the future. An end-user is used this system on an OS; either it is Windows or Linux. The System shall run on PC, Laptops and PDA. etc. The technology should be transferable to different environments easily.

**8.6 Accessibility:**

Only registered users should be allowed to access the next process after authentications. Only GUI access of the system should be permitted to end users.

**8.7 Policies:**

The system should adhere to all the legal formalities of the particular countries. The system should maintain security related to sensitive data.

**8.8 Efficiency:**

The system should provide good throughput and response to multiple users without burdening the system by using appropriate number of servers.

**8.9 Safety:**

Software should not harm ethical and environmental conditions of the end user's machine.

**8.10 Modularity:**

The system should have user friendly interface. It should be easily updated, modified and reused.

**9 Operational Scenario:**

**9.1 User Interaction**

user can search yoga center and yoga trainer in their city. User can learn basics of yoga. The system shows all yoga courses and information about yoga teachers to end users. User can choose class and also teacher. If user wants to cancel the request before selecting the course then he or she can cancel it. User will receive email about all information he or she select the class and also schedule of the class and user can contact directly to the yoga teacher for knowing the extra information about the class but after the selects course.

**9.2 Yoga teacher Interaction:**

**9.3 Staff Interaction:**

BOD (Board of Directors)

**10 Preliminary Schedule:**

1.Login

2.Manage user database Browse category

3.choose courses and yoga teachers

3.add or remove courses from the page and also update the information about yoga teacher

4.Manage user database

5.Manage yoga teachers and courses’ schedules

6.update courses

7.select/reject courses

8.Logout

9.Give feedback

10.payment (Donation)

11.ByCreditCart By Debit Card By online banking

12.Visit Site

13.Create new account

14.View account details

15.Delete account

16.Registration

17.User Support