Online Sports Management System

Software Architecture Document

Version 1.0

**Revision History**

|  |  |  |  |
| --- | --- | --- | --- |
| **Date** | **Version** | **Description** | **Author** |
| 07 May 2014 | 1.0 | Initial Software Architecture Document for Online Sports Management System for UDEL CPEG612 Spring 2014 Team #1 | Jason McGlamary |
|  |  |  |  |

**Team #1 Members:** Jason McGlamary, Dan DeCamillo, Alice Guerrieri

**Software Architecture Document**

1. **Introduction**
   1. Purpose

This document is meant to provide a comprehensive overview of the software architecture for the proposed system “Online Sports Management System (OSMS)”. By utilizing various different architectural views, the aspects and components of the system will be made clear. Only significant architectural decisions on the system are conveyed.

* 1. Scope

This Software Architecture Document provides a functional overview of the OSMS.

* 1. Included Graphics

Full-sized files containing the diagrams used in this document are provided to aid in visibility. Please refer to the “Diagrams” folder included with the distribution of this document.

1. **Architectural Representation**

This document presents the architecture as a series of views including use-case view, deployment view, dataflow view, and model control, and database view. Most views utilize or extend a form of the Unified Modeling Language (UML).

1. **Architectural Goals and Constraints**

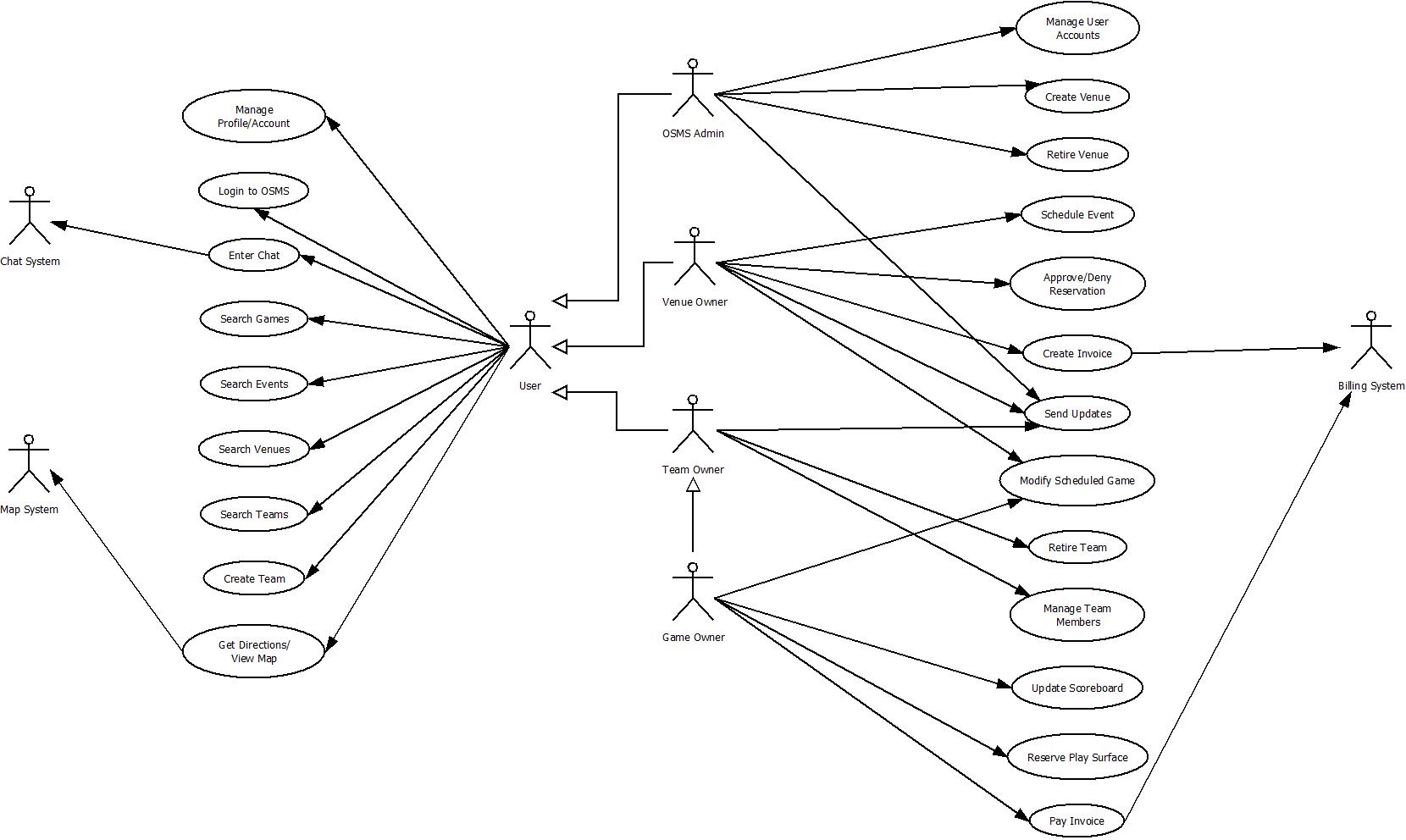
Taken from the Software Requirements document, several key requirements and constraints were identified to significantly impact the architecture.

* + The System must facilitate searching for Venues, Teams, and Games according to proximity to a user-selected area.
  + The System must facilitate scheduling Games at available Venues by requesting Teams composed of Individual Users of the System.
  + The System must be accessible through standard web browser over the Internet.
  + The System must provide directions to scheduled games.
  + The System must provide a billing mechanism for Venues.
  + System Users’ profiles must allow storing of user preferences.
  + The System must automatically logout users after a period of inactivity.
  + The System must log User activity within the System.
  + The System must provide a Chat functionality to allow communication between Users.
  + The System must accommodate changing and expanding types of Games.

1. **Use-Case View**

The use-case view describes the set of scenarios and/or use cases that represent some significant, central functionality. It also describes the set of scenarios and/or use cases that have a substantial architectural coverage (that exercise many architectural elements) or that stress or illustrate a specific, delicate point of the architecture.

* 1. Significant OSMS Use Cases:

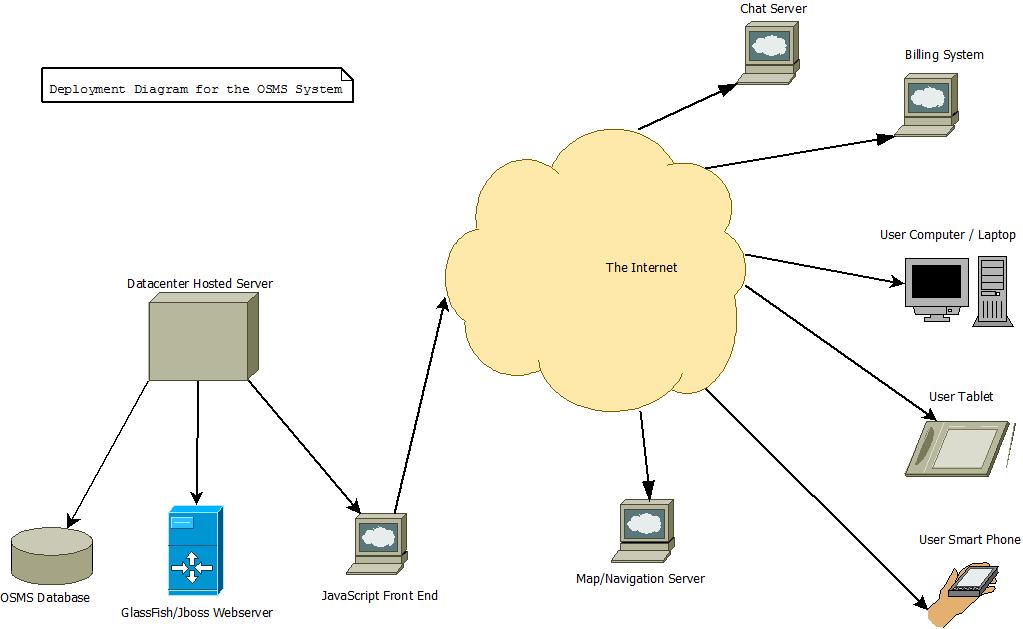


* + 1. Login to OSMS – describes how a user logs in to the OSMS application. The actors for this use case include all descendants of the User Actor including OSMS Admin, Team Owner, and Venue Owner.
    2. Manage Profile/Account – describes the act of a User maintaining his/her own profile information including contact information, account password, and closing account.
    3. Enter Chat – describes the User entering into a communication chain with one or more other Users. Can be used to chat, or to assist coordinating other use cases in the system such as requesting to Join a Team.
    4. Search Games – describes a User searching for Games which have already been scheduled within the system.
    5. Search Events – describes a User searching for Events which have already been scheduled within the system. Events may include Tournaments or other activates which may make normal Game scheduling impossible for a given time period at a certain Venue.
    6. Search Venues – describes a User searching for Venues within a specified geographic area.
    7. Search Teams – describes a User searching for Teams created within the System. The User may simply be looking for Teams which are playing within a specified geographic area or may be searching for Teams which are accepting recruits with the intention of joining the Team.
    8. Create Team – describes a User electing to start his/her own New team and thereby the User becomes the Team Owner of that new Team.
    9. Get Directions/View Map – describes a User requesting travel directions from a specified location to a chosen Venue.
    10. Manage User Accounts – describes the OSMS Administrator actor maintaining the accounts of the OSMS system and includes password resets, and User account deletion and deactivation.
    11. Create Venue – describes the OSMS Administrator creating a new Venue within OSMS which will then be available to schedule Games and Events.
    12. Retire Venue – describes the OSMS Administrator disabling a Venue within OSMS thereby making the Venue unavailable for scheduling new Games and Events.
    13. Schedule Event – describes the Venue Owner Actor reserving his/her owned venue for a specified date or date range for a Special Event thereby preventing Teams from reserving the Venue on those dates.
    14. Approve/Deny Reservation – describes the Venue owner approving or denying a Team Play Surface Reservation Request. This approval can be automated at the discretion of the Venue owner.
    15. Create Invoice – describes the Venue owner electing to bill a Team for reservation and use of a Venue Play Surface.
    16. Send Updates – describes the case of a Team Owner, Venue Owner, or OSMS Admin to send informational updates to appropriate users.
    17. Modify Scheduled Game – describes a Game Owner or Venue Owner rescheduling or cancelling an existing Game.
    18. Retire Team – describes a Team Owner electing to disable his/her owned Team effectively removing it from OSMS.
    19. Manage Team Members – describes a Team Owner adding and removing Users to his/her Team roster.
    20. Update Scoreboard – describes the Game Owner of a scheduled Game modifying the game day scoreboard during Game play.
    21. Reserve Play Surface – describes a Team Owner requesting a Play Surface Reservation at a specified Venue to play a Game. This action makes the Team Owner the Game Owner for this Game and Game Reservation.
    22. Pay Invoice – describes a Team Owner remitting payment to a Venue via the Billing System for the reservation of a Play Surface.

1. **Deployment View**

The Deployment View describes the various physical nodes for expected platform configuration of OSMS.

* 1. Deployment Details

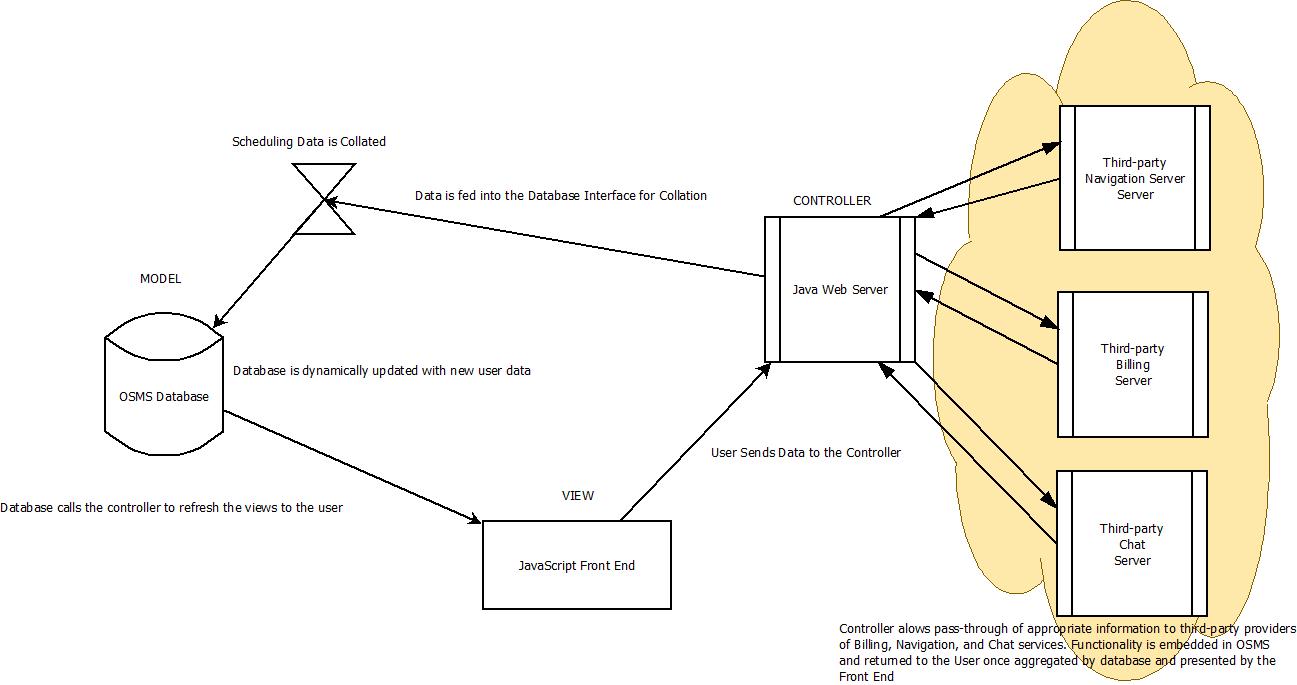


* + 1. Datacenter Hosted Server – the OSMS system is to be hosted and served by a third-party Web Hosting Service allowing flexibility and scaling while making the system easier to maintain and support for the client who will not have to provide dedicated resources to host the system.
    2. OSMS Database – the backend database, likely mySQL or similar, which contains all of the data generated by OSMS. Transactional logging is possible which will fulfill the requirement for the System.
    3. JavaScript Front End – a JavaScript Front end which is responsible for presenting the graphical interface of OSMS to the User.
    4. Chat Server – the Chat functionality for OSMS will be provided by existing third-party vendor agreement. Again, this should reduce support costs compared to implementing a home-grown solution.
    5. Map/Navigation Server – the Direction/Navigation functionality of OSMS will be provided by an existing third-party vendor product, likely Google Maps, which can be easily incorporated into OSMS through established APIs.
    6. Billing System – the Billing functionality of OSMS will be provided by an existing third-party vendor product, likely PayPal, which can be easily incorporated into OSMS through established APIs.
    7. Personal Computer – Users access OSMS by connecting over the Internet using Personal Computers including Desktops and Laptop devices.
    8. Table Computer – Users access OSMS by connecting over the Internet using Tablet Devices running supported Android and iOS operating systems.
    9. Smart Phones – Users access OSMS by connecting over the Internet using Smart Phones running supported Android and iOS operating systems.

1. **Dataflow View**

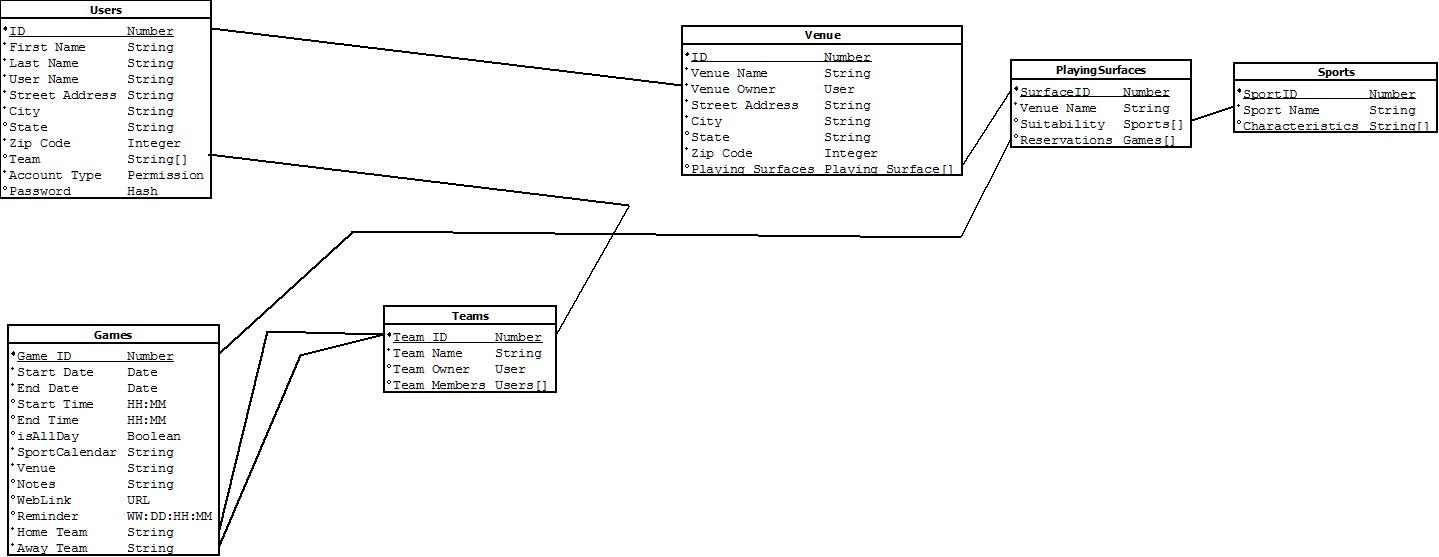
The Dataflow View presents the passing of data as it traverses the System. In the case of OSMS, additional functionalities are provided by third-party vendor products over the Internet. Only the necessary information relating to each service is communicated via embedded API. Data related to the scheduling aspect of OSMS remain within the OSMS system only.

* 1. Dataflow Details



1. **Database View**

The Database view serves to illustrate the possible structure of the critical Tables in the OSMS database. The relationships between these tables are also indicated. Here, only the critical objects necessary to schedule a Game or Event are shown.

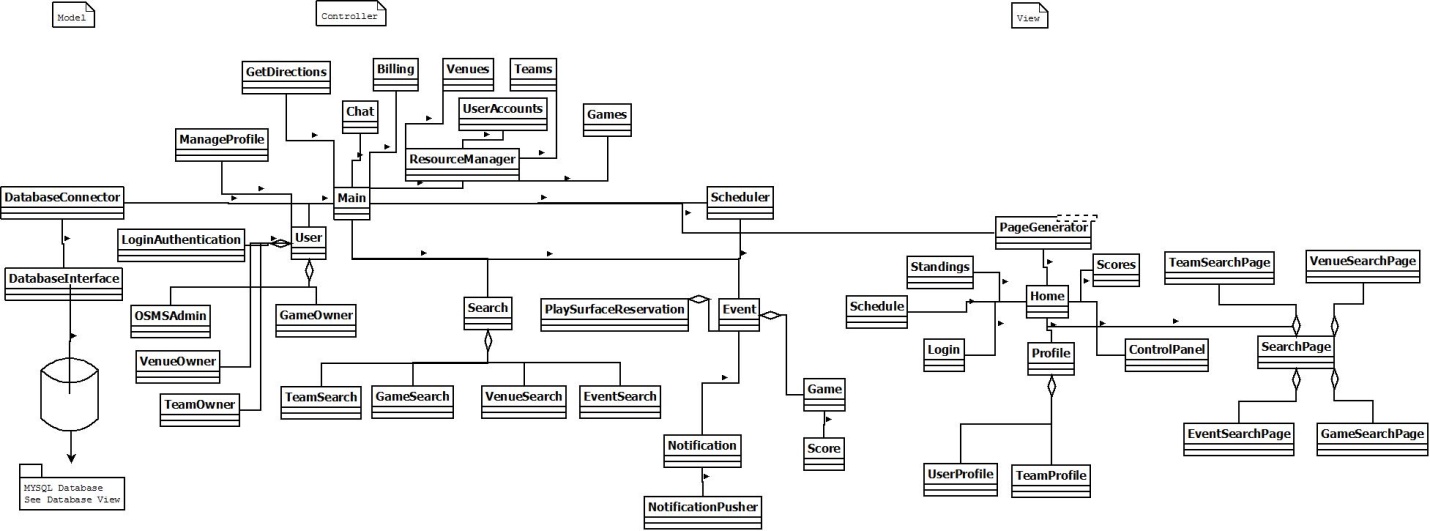


* + 1. Users – The Users table illustrates the properties of a User. It contains appropriate account and contact information. A User serves as the primary building block in creating a Team Roster. A User entry is also used to specify the owner of a Team object and a Venue Object.
    2. Teams – The Teams table shows the properties of a Team. Mainly, each Team has a Team Name and one or more Team Members of type Users.
    3. Venue – The Venue table shows the properties of a Venue. Each Venue possesses a Name, Address information to provide its location, a Venue Owner, and one or more Playing Surfaces which may be reserved for events in OSMS.
    4. PlayingSurfaces – The PlayingSurfaces table shows the properties of a Playing Surface. Playing Surfaces are associated with the Venue where the Playing Surface is located. Additionally, each Playing Surface contains a listing of Suitabilities indicating which Sports that particular Playing Surface may be used for. Finally, a Playing Surface contains the Reservations which have been made for that Playing Surface. Each Reservation is an individual Game.
    5. Sports – The Sports table lists the type of sports which OSMS is able to schedule. It includes the sport Name as well as any Characteristics that are properties which may better describe each particular sport.
    6. Games – The Games table shows the properties of a Game. Mainly, it aggregates information from the other Tables to create a Game reservation. Date and Time information are combined with Venue location and competing Teams to create the Game. Additionally, fields like Reminder, Notes, and WebLink extend the usability of OSMS by providing additional details to be communicated to Users about an individual Game.

1. **Model View Controller and Concept Diagram**

The Model View Controller architecture separates an application into 3 interconnected parts. This separation delineates internal representations of the System into an application or business data model, the View which is presented to the End-user, and the Controller which is responsible for accepting input and converting that input into commands which update the Model or View.

* 1. Model View Controller Details:



* + 1. DatabaseConnector – Provides the assembly point for updates to and retrievals from the System database. The Database Connector provides optimization for database command queuing and ordering. Actions to be performed on the Database and its data model are received from the Main controller. Data received from the Database is received by the DatabaseConnector and distributed to the Main controller and from there sent either to the Scheduler Controller or to the PageGenerator to be viewed by a User.
    2. Main – The Main Controller is responsible for manipulating data to make it useful to either the Database or to the End User. The Main Controller handles the following functionalities of OSMS:
  + Management of System resources including User accounts, Venues, Teams, Games.
  + Coordination of Chat System
  + Coordination of Billing System
  + Coordination of Navigational System
  + User Authentication and Login
  + Search functions
  + Invocation of the System Scheduler Controller
    1. Scheduler – The Scheduler Controller is responsible for taking data from the Database through the Main Controller and combining it with data supplied by the View Controller from the End User to create and schedule Games and Events. It is through the Scheduler that PlaySurface reservations are created, Game Scores are updated, and Notifications are sent to End Users.
    2. PageGenerator – The PageGenerator formats all data for display to and interaction with the End User. Processed data from the Main Controller is optimized for presentation to the User and informational updates from the End User is processed and returned to the Main Controller for further processing or to update the Database.

1. **Size and Performance**

The OSMS System is intended primarily for use by neighborhood intramural teams of Users. As such, initial utilization and penetration is expected to be very small as the user base slowly grows through word of mouth and advertisement to area Venues. However, over time, the System has the potential to be used nationwide, and could be extended even further, if desired. By abstracting the hosting portion to a vendor provider with a proven track record in web service hosting, it can be certain that the application remains scalable and responsive over time, yet protecting the client from over-allocating resources at any given time.

1. **Quality**

Again, abstracting the hosting services and Billing, Chat, and Navigation components to existing providers with known levels of service quality goes a long way to ensure the overall quality of OSMS as a whole. Since there is no attempt to reinvent services which are already known good, the client does not have to individually maintain these complex components in the implementation of OSMS. Beyond these factors, the architecture supports the following additional quality requirements:

* + The user interface must be intuitive and require no additional training to use.
  + The System shall provide built-in online help for the User.