Software Requirements Specification
Document Outline

1. Introduction

The system is a 16-team single elimination tournament bracket management system.

1.1. Problem Statement

The Bracket Management System allows for an easy way for a fair (400+) number of users to save their bracket selections and to be automatically scored and compared to other users.

1.2. System Personnel

a. Users

People that intend on filling out a bracket to be scored by the system.

b. Administrators

Most likely the customer that buys the product. Is responsible for the installation and entry of the setup data for the tournament brackets as well as maintaining the server that that product will be running on.

c. Project Manager
d. Developers

1.3. Operational Setting

a. Server Requirements
    a. A web server capable of running Java Servlets
    b. A SQL server to provide data storage on quick retrieval on the backend.
b. Client Requirements
    a. An internet connection
    b. A modern browser with adequate javascript and CSS support (IE 7+, FireFox 2+, etc.)

1.4. Impact Analysis

Ideally a piece of bracket management software could fairly and easily manage scoring for an office's betting pool for an NCAA tournament, NFL Playoffs or any other sporting event. This could decrease the time that employees spend in non-work related activities such as managing such a betting pool, while increasing the time employees might spend at a work place viewing and managing their brackets. Depending on whether or not this software is being sold to an employer it may or may not be an issue.

Even though most of the code will be running on the client's web browsers, which while require them to be running software that is of a reasonable age (no more then five years old) to the administrator the most important impact will be on the server. Java Servlets have the potential of taking up a considerable number of CPU cycles on the server due to inefficiencies in the Java Virtual Machine and general Java overhead issues and could hurt the scalability of the product. While for the current user number target (400) this should not be a problem.

1.5. Related Systems

Related systems are those computer systems in use elsewhere that provide functionality similar or related to the functionality of the system being specified here. Both commercially available and public domain systems.

1.5.1. Related System A

Facebook recently added a large bracket management system in preperation for the NCAA Men's Basketball Tournament that was tightly integrated with their website.

a. Integrated support in a large community network
b. Viewing other people's brackets
c. Automatic scoring
d. Prominent display on many facebook pages
e. Game scores and detailed pages on the games, while they were there, were difficult to find.
f. Support for smaller "pools" with groups of friends

2. Functional Requirements

Functional requirements define the specific functions that the software system performs, along with the data operated on by the functions. The functional requirements are presented in scenarios that depict an operational system from the perspective of its end users. Included are one or more examples of all system features and an enumeration of all the specific requirements associated with these features.

2.1. User Interface Overview

The user interface is broken down into three basic areas. A User Login and Registration area, the User Bracket Selection Area and the Administration Area. Both the User Bracket Selection Area and Administration Area must be accessed through a login first.

A user login and password page will be the first thing seen. It will include fields for entering the user name and password along with a link to a page to register for a new account.

2.2. *System-Specific Requirements*

2.2.1 *User Login and Registration*

- Allow users to login with username and password (1a)
- Users should be able to login with any internet accessible computer (1b)
- Users must log in to be able to modify and save a bracket.

2.2.1.1 *Login*

- Provide for a user name or password
- If the login is incorrect error appropriate "Could not find User Name and/or Password" and show the registration page.

2.2.1.2 *Registration*

- Must have a "create a new account" page
- Can ask for an email, but don't do anything with it beyond saving it in the database.

2.2.2 *User Area*

This is the area that a normal user would see upon logging in correctly. It should default to the bracket selection page.

2.2.2.1 *Bracket Selection Page*

- Display a 16-team single elimination bracket (1c)
- Load the user's previously saved bracket selections on login
- Allow selections on all games including the championship. (1d)
- A "save" button for storing changes (1e)
- Changes do not get saved until the user hits the "save" button
- Locations should be displayed for all games (1c)
- Stats such as scoring records should be available ?
- Highlight incorrect and correct choices once winners have become known. (1k)
- User's score should be displayed (1k)
- Do not allow changes one it has been locked (1j)
- No other users scores on this page.
- No storing multiple brackets per user.
- Brackets do not need to be complete to be saved.

2.2.2.2 *Bracket Score Rankings Page*

- A listing of other users scores should be available on a separate page from highest to lowest. (1l)
- Can be done in specified numbers at a time (ie 25 at a time instead of all users)
- Users can not view other users' brackets

2.2.3 *Administration Pages*

This is the area that the Administrator uses to perform tasks necessary for maintaining the bracket once it is installed.

- There is only a single admin user that administrates the brackets, but does not save a bracket themselves.

2.2.3.1 *Bracket Setup and Edit Page*

- Allow for input of the first round teams *somehow*
- Allow for input for the locations and start times for all the games
- Allows for changes to the first round team names
    - Editting names becomes inaccessible once a bracket has been saved

    - It may be necessary to have a feature where the admin locks users out of making brackets.
- Allow for changing the game locations and start times
    - Locking functionality on this is unspecified.

2.2.3.2 *Lock Page*

This page consists of a description of the functionality of the lock button and the lock button itself.

- Prevents any future changes to users brackets.
- Brackets do not get locked until the admin explicitly specifies it.

2.2.3.3 *Winning Bracket Page*

Displays a bracket similar to the one that users use to set their selections, but is used to designate the winners for each round.

- Entry/Selection of winning teams on a bracket. (1h)
    - Save button on winners bracket?

2.2.3.3 *Scoring Mode Selection Page*

There needs to be implemented at least two different scoring modes for the bracket system. Simple and one other of our choosing.

- Simple (2a/1j)
- Increasing value by one for each round (2b)
- Upset-biased (simple + extra point for every correct prediction when a lower ranked team wins and difference in ranking > 2) (2c)

2.2.3.3 "*Create a new tournament*" *Page*

A page that describes the function of the button warning the administrator of it's exact functionality and the button itself.

- A Clear All or "Create a new tournament" option that erases the current tournament and all saved user brackets

3. Non-Functional Requirements

Non-functional requirements address aspects of the system other than the specific functions it performs. These aspects include system performance, costs, and such general system characteristics as reliability, security, and portability. The non-functional requirements also address aspects of the system development process and operational personnel.

3.1. System-Related Non-Functional Requirements

Non-functional system requirements include some or all of the following:

a. performance

- i. time
  - a. It should not consume more then 5% of the CPU cycles on a reasonably fast server.
- ii. space
  - a. It should not require more then 200 MB on a web server, and negligible amounts of space in a user's browser cache.
  - b. Each individual page should not be more then 20 KB to ensure reasonably fast load times.

b. operational environment

- i. hardware platform
  - a. Modern Server Hardware (x86, Opteron, Sparc, etc.)
- ii. software platform
  - a. Operating Systems (Linux, OS X, Windows) capable of running Apache Server and Java Servlets
- iii. external software interoperability
  - a. Apache Tomcat

c. standards conformance

- i. HTML4
- ii. CSS
- iii. Java Servlets

d. general characteristics

- i. reliability
  - a. The code should run reliability and not create any issues with common browsers or the server it runs on.
- ii. robustness
  - a. It should be able to support small (up to 400) numbers of users without noticeable slowness or any errors.
- iii. accuracy of data
  - a. User inputted data (brackets) should be 100% accurate according to what they saved
  - b. Tournament data is reliant on an administrator accurately entering the data, there is no system for ensuring the Administrator enters it correctly.
- iv. correctness
  - a. Scoring needs to be 100% correct based on the data given by the admin.
- v. security
  - a. While the product has no built in security features, it should be easily integrated with a SSL server for logins, if needed.
- vi. privacy
  - a. User Brackets are not viewable by other users
  - b. User emails are only viewable by the admin
- vii. safety
  - a. There should be minimal safety repercussions from use of the product.
- viii. portability
  - a. The client code should be highly portable between various modern browsers (Internet Explorer, Firefox)
  - b. The server code however is very dependent on both a SQL server and Java Servlets.
- ix. modifiability and extensibility
  - a. There is no guarantee that the product will be easily extendable beyond what is provided in the spec.
- x. simplicity versus power
  - a. The average user's interface should be very simplistic with few options beyond the tournament choices themselves
  - b. The admin should have the option of setting various scoring rules that effect all the users, but will have no ability to customize beyond any preset rules already in place.

3.2. Process-Related Non-Functional Requirements

Non-functional process requirements include some or all of the following:

a. development time
b. development cost
c. software life cycle constraints
d. system delivery
    i. extent of deliverables
    ii. deliverable formats
        a. HTML and CSS pages to run place on the server and run on client browsers
        b. Java Servlets to run on the server to properly feed data to the clients.
e. installation
    i. developer access to installed environment
    ii. phase-in procedures to replace existing system
f. standards conformance
g. reporting
h. marketing
    i. pricing
        a. Pricing has not been specified
    ii. target customer base
        a. College students
        b. Office Workers
i. contractual requirements and other legal issues
    i. Have to avoid using trademarks (ie. NCAA) that we have no legal right to use.

3.3. Personnel-Related Non-Functional Requirements

Non-functional personnel requirements include some or all of the following:

a. for developers:
    i. credentials
    ii. applicable licensing, certification
b. for users:
    i. skill levels
        a. Should know how to use a mouse and keyboard
        b. Should be able to use a web browser
        c. Have a vague familiarity with how a tournament bracket works.
    ii. special accessibility needs
        a. No inherent support for screen reading technology or similar assistance for the blind is planned.
    iii. training
        a. No training should be needed or is planned to be provided.

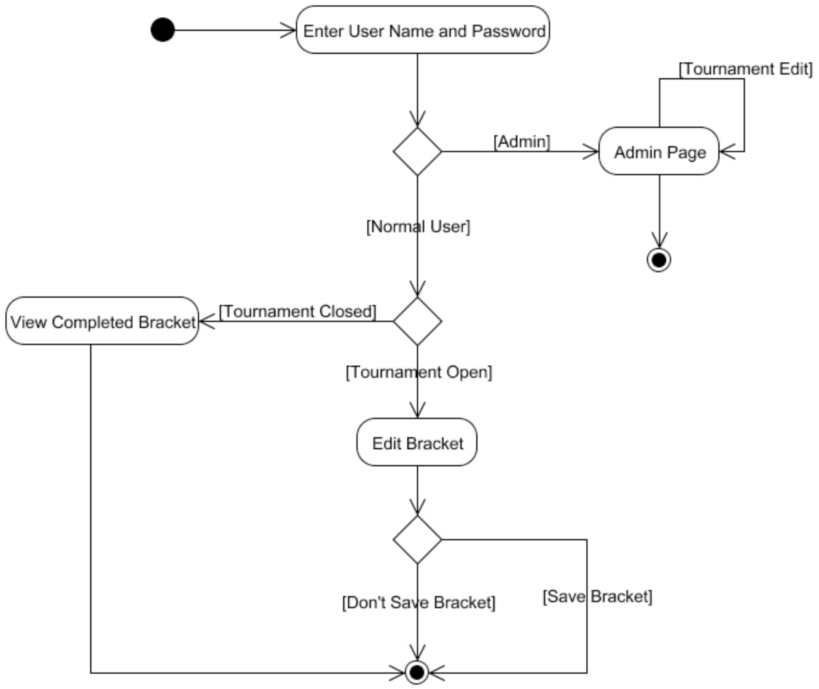
4. Developer Overview

4.1. System Design Executive Summary

The system will be contained in three major objects: server code, client code, and a data storage file. The server code will handle login processing, bracket manipulation, and admin functionality. This backend code will be unseen by the users and does not involve any GUI components. The client code is the control interface where the user can manipulate their bracket and view their score once a tournament has begun.

The administrator screens which allows the admin to modify or start a new tournament are include in the client code as well. The data storage file contains the users saved brackets along with username and password information. This tab delimited file will concisely store all necessary information associate with a user.

4.2. A UML Activity Diagram of the User's Experience



Contributed by Dr. Gene Fisher
Adapted by Dr. David Janzen 4/4/07
Edited by Ryan Pangrle 4/9/07