Phase 2

Website Implementation Specifications

People with Purpose

www.peoplewithpurpose.mooo.com

Mohammad Bashir 996196030

Group: The Moe

Table of Contents

Features and Functionality Specification.. 2

Section Descriptions.. 3

User 3

Social... 3

Environment.. 4

Administrative.. 4

Use Case Diagram.. 5

Project Plan.. 6

Team Organization... 6

Development Schedule.. 6-7

Software Architecture and High-level Design... 8

Overview.. 8

Functional Module Diagram... 8

User Specific Module Diagram of Engine................................. 9

Administrator Specific Module Diagram of Engine...................... 9

Data Representation... 10

ER-Diagram... 10

Schema... 11

Clarifications... 11

Testing Strategy and Plan.. 12

Overview... 12

Test Schedule.. 12

Changes.. 13

Palette.. 14

Features and Functionality Specification

In an attempt to construct a sound foundation for a powerful conceptualization of community of practice, a model based heavily on descriptive qualities has been constructed. Certain words that append themselves to the image, accessibility and feel of the working website are the main directional factor in development. Our “People in Action” website will be described as simple, bold, adaptive, personal, and will build cliques of avid contributors while alienating users that threaten to poison the community. While some of the descriptions may seem aggressive, they are meant to attract the right user. The following rundown and UML diagram both describe the main portions of the website and how our key words fit into a system intending to obtain a strong and loyal community of satisfied users.

Development focused on users has been broken into three sections that all pertain to separate parts of the website, but all will obtain and contain information that is intended to weave the site together. The three sections of the site will be referred to as “User”, “Social”, and “Environment”. Along with the three main sections of the website there will be administrative access which will incorporate a upper-level control of the website without forcing administrators to alter server-side scripts.

Boldness will be implemented in a thorough and complete design that will focus on an intriguing, yet inviting color scheme. Buttons, banners, bars, and other interaction tools will be custom designed with scroll-over reactions. The goal of applying attractive simplicity to the website is to “wow” users, and generate a professional atmosphere. Color pallet marked figure 1 is the potential color schemes for the website.

Section Descriptions

User

- Account
  - Users will have access to a page displaying buddies, personal information, recent events, past events, and pending requirements for various responsibilities such as leaving a reputation feedback for other users. Accounts will be simplified for ease of access, to create a comfortable control system for a user.
- Profiles
  - Users will have the opportunity update personal profiles with relevant information pertaining to their occupation, specializations, personal life etc. The profile page will be built to sell the user's image to others to create voluntary social inclusion within the community of users already built. User profiles are where the term personal becomes relevant.
- User Ranking
  - Stated separately, but seamlessly woven into any personal and interactive aspect of the website, a global ranking system will decide a users contributory worth. Calculated non-linearly from other systems of user valuation on the website, User Ranking will be an accumulation of feedback received by an account. A user having numerous negative complaints, too many low ranked events etc., will have a lower User Rank. A User Rank will not create a level system for Users, but will simply calculate the consistency of a user. Someone with one event with good ratings will be just as valued as someone with 10 events with good ratings.

Social

- Buddy System
  - A system to encourage expanding one's network and knit the community into a tighter state. There will be no direct allowance for users to add others as friends. Instead, upon joining an event, if one user is to give another user a good reputation score then that person will automatically join their buddy list. People on a users buddy list will be more likely to make appearances in searches and listings.
- Reputation System
  - Users will be able to provide feedback ratings for events created by other users, these scores will be visible on said activities, on profile pages, and will be used towards User Rank calculations.
- Event Registration
  - A user will be able to create and cancel an event. There will be restrictions set by administrators for the deadline to cancel an event. A cancellation deadline will avoid

providing users with the ability to cancel events too close to when the event is to be activated. Events will be created with user entered keywords that will be search criteria.

- Event Participation
  - A user will be able to join events. When an activity has met its requirements, it will become active and all participants will be locked to it to be rated regardless of if they attend or not.

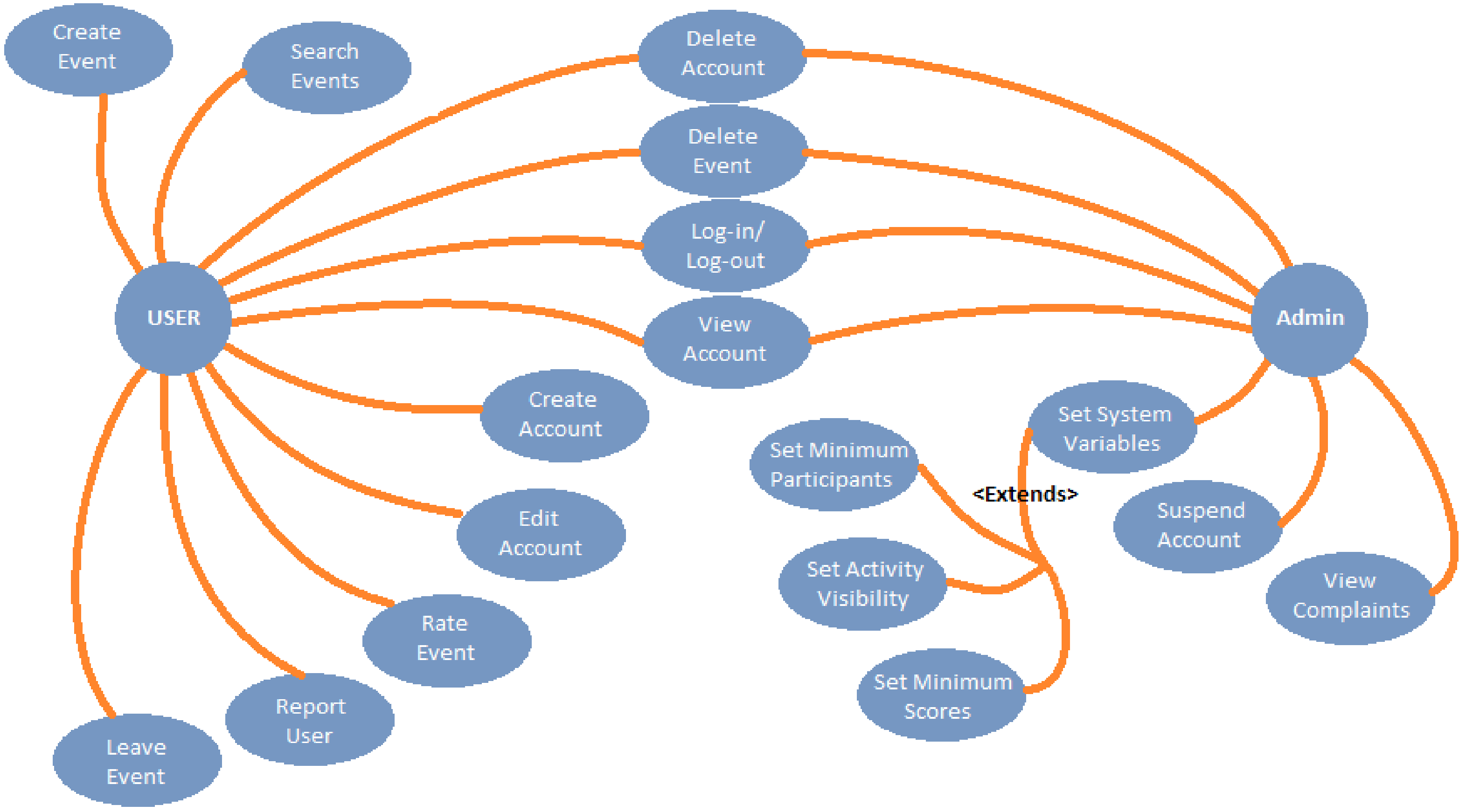
Environment

- Account Registration
  - To access any information given to and provided from the website community, a user will have to register an account with the system providing information like an email address for validation.
- Authentication
  - Log-in and log-out systems will be provided to users. Passwords will be controlled by password hashing, and information of passwords will not be given to administration.
- Complaints
  - A user will have a section of the website to complain about system flaws, aesthetic and functional issues, and other users. Complaints may affect a user's User Ranking, and the infrastructure of the website.
- Search/Listing Constraints
  - The terms clique and alienating have the most power in this aspect of the website. The search and listing system will adopt filtration that can trim lower scored users and keep higher ranked users at the top of lists. The adaptive nature of the search engine takes into account the contribution and behavior of a user in the community.

Administration

- Usage Statistics
  - Administrators will be able to view nominal information about users, events, ratings, rankings, and complaints to monitor the health of the community thus make alterations in the website system accordingly.
- System Management
  - Administrators will be able to set requirement variables for various sections of a website, such as the max difference between users to be visible in search listings, minimum joins for an event to become active. An administrator will be able to set minimum scores for an individual to create an event and alter other constraint data.
- Accounts Management
  - Administrators will be able to delete, suspend, or warn accounts about behaviors
- Activity Management
  - Administrators will be able delete, or alter information about activities

User/Administrator and System Interaction Use Case Diagram



Project Plan

Team Organization

Currently, the development team consists of a single team member, and under such team circumstances there is a single reign over all issues. Responsibilities required to complete the project will be organized in a fashion to sustain functionality in the website while building on the system. All code will be under version control using Source forge. Although keeping code online is not necessary, it will assist with reverting versions of software, if complications arise, and organization.

Development Schedule

The organization of the development schedule has been constructed to account for dependencies. One milestone will not find entire completion without the previous milestone being finished. The heaviest requirements fall on the first milestone. The configuration of server software and completion of the database is required and necessary to implement any other part of the system and for testing.

Project Schedule

		days
• Obtain and configure all relevant dependency software and obtain URL	COMPLETE COMPLETE	1
• Create and deploy all relevant SQL tables	COMPLETE	2
• Test data entry and retrieval	COMPLETE	1
Functional and Linked Website		4
• Create user account creation system	COMPLETE	3
• Create user authentication system	COMPLETE	3
• Create user complaint system	UNDER CONSTRUCTION	1
• Create search engine		3
• Test account registration, logging systems, and complaint tracking		1
Complete Environment System		11
• Create User Rating system		2
• Create account view page		1
• Create profile edit page		4
• Test User Ratings on created pages, and profile editing		2
Complete User System		9

• Create rating system	4
• Create buddy system	4
• Create event registration/participation pages	2
• Test user rating and auto addition of buddies, test event creation deletion and activation	2
Complete Social System	12

• Create server-side data logging	2
• Create data display system	1
• Create Accounts Management page	3
• Create Event Management page	3
• Test data logging accuracy, and global variable adjustment robustness	3
Complete Administrator System	12

• Create/edit all relevant page art	3
• Implement all relevant page art	2
• Test coordinates and placement of art, loop to beginning of milestone task until completion	2
Complete UI Implementation	7

Total Approximate Days For Implementation	55

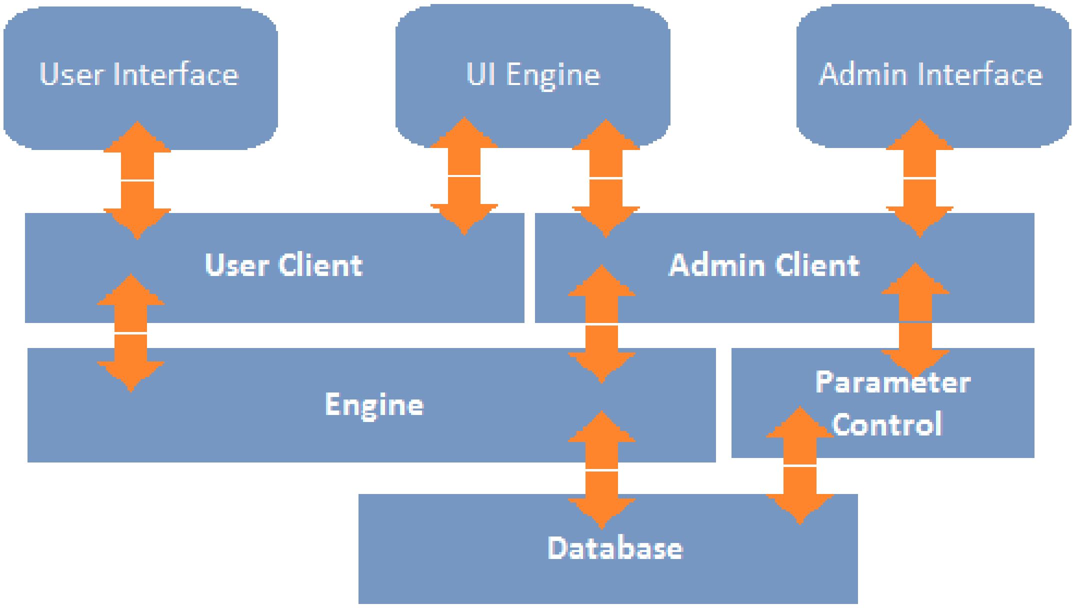
*Approximation has been calculated solely from summation, thus discounting the reality of certain tasks requiring less than a days work. Stated total implementation time realistically should be less considering certain factors such as most tasks stated as 1 day will most probably take far less than a day to complete and multiple tasks may be addresses the same day.

Software Architecture and High-level Design

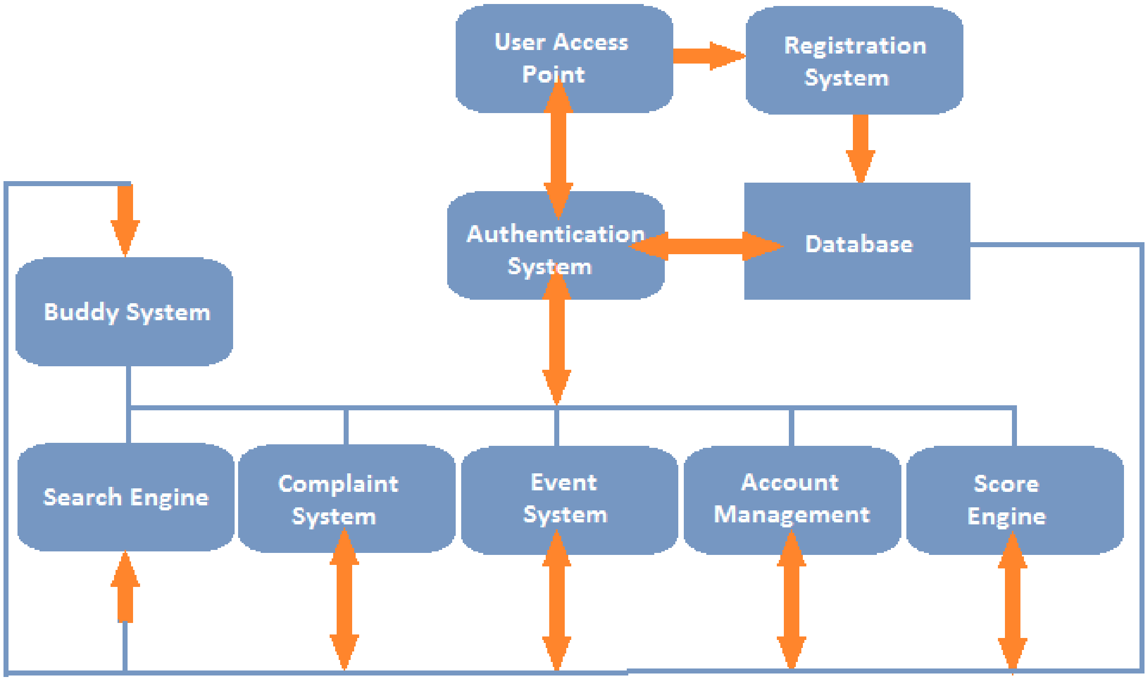
The planned infrastructure of the website and all of its inclusive dependencies will be built on LAMP. LAMP, a package of synergistic web-development software including Linux, Apache web-server, MySQL, and PHP, will mark the main modular distinctions between pieces of the website. The Linux operating system will be the carrier of all server activities, the web tree, and data. Apache web-server will be the server module connecting all server activities to user activities. MySQL will be the system of data querying and tracking. All user and website information, static and dynamic, will be contained in tables constructed, referenced, and altered by a database in MySQL. PHP will be the backbone of all server activities. All data retrieval/placement, server-end validation, authentication, and server security will be programmed into an engine using PHP. The user interface for client-end will be built dynamically, constantly referring to JQuery scripts. Users will explore a website with drop down menu lists, mouse-over button effects, and other attractive features that will keep the web user interface fresh and fluent.

An upper level perspective of the most general framework of the website is shown below. The UI Engine will maintain all dynamic content of the website. User Client and Administrator clients, although very similar, will maintain some strict and essential differences. Administrators will have access to control sections for variables that will behave as constraints to a number of interactions a user may have with the system, as well as sovereign control on certain dynamic data that may exist in the database. The main engine will maintain all of the functional features of the website and will be accessed by both administrators and users.

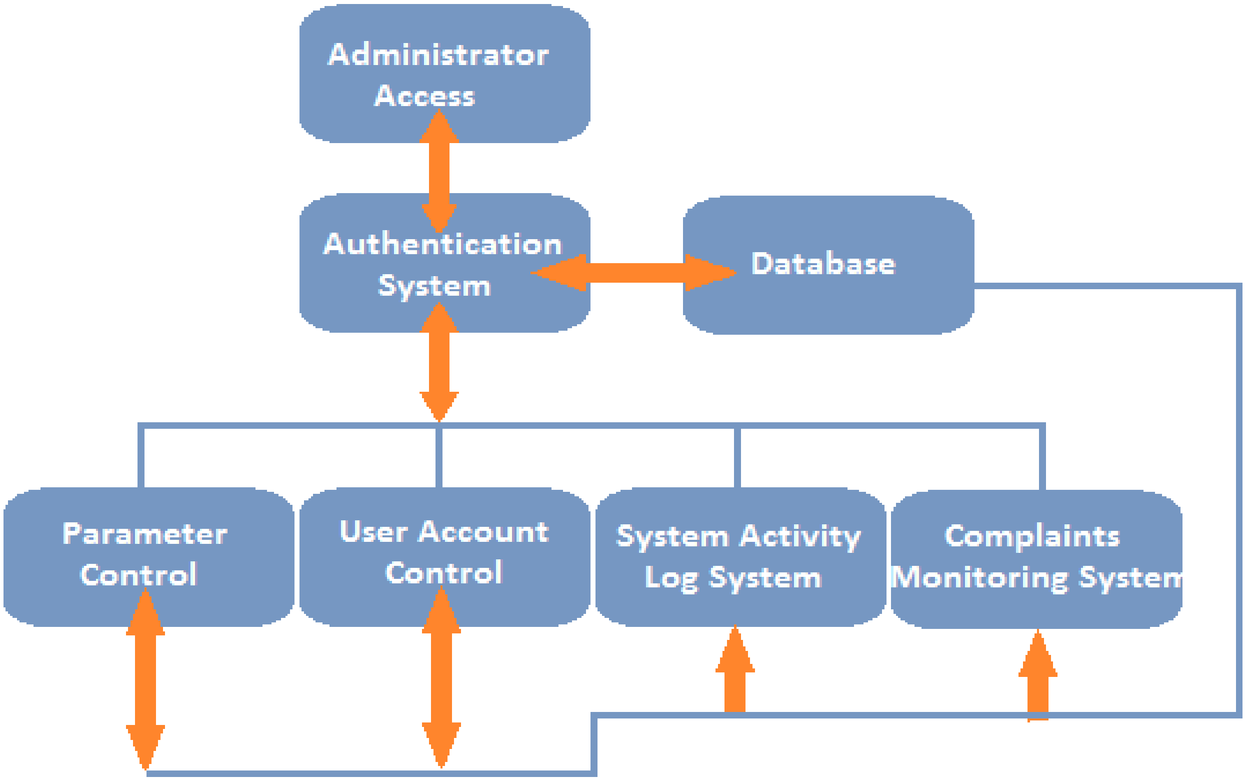
Functional Module Diagram



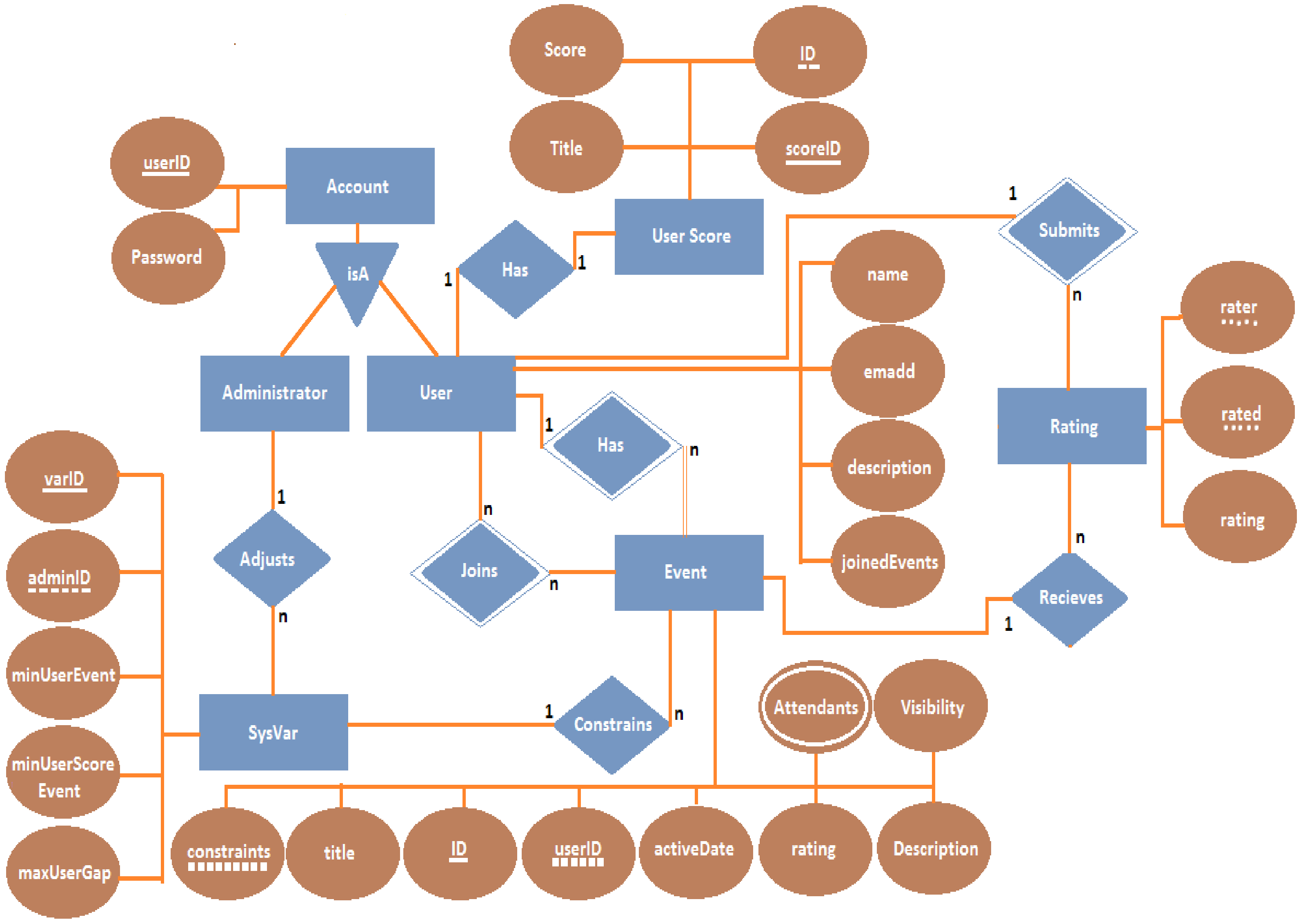
User Specific Module Diagram of Engine



Administrator Specific Module Diagram of Engine

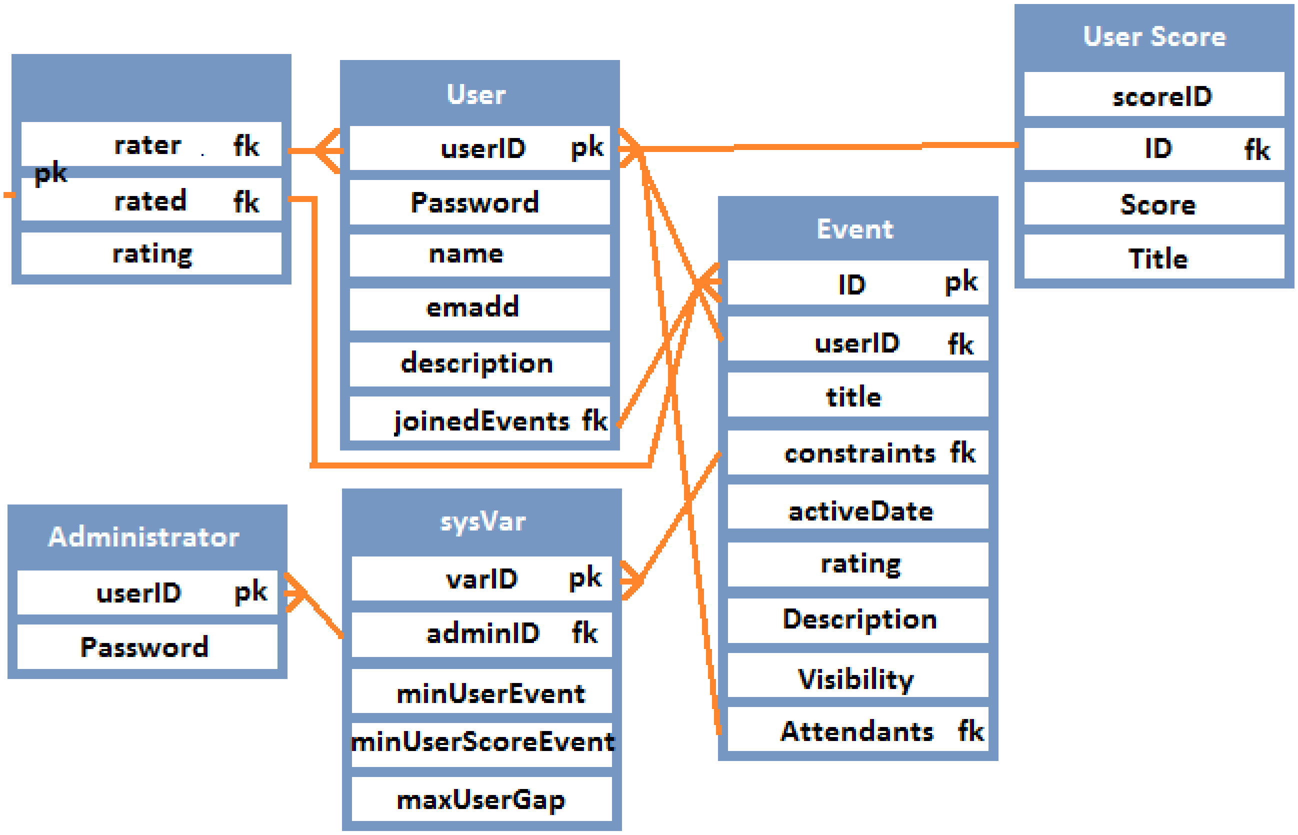


Data Representation



Data Representation

Schema



Clarifications

Data tables with the description value are actually entities that require a mass amount of variable information that may or may not exist depending on conditions that a user account may be in. The description column is actually a file path to a restricted access XML file that will contain the information required for certain accounts. An example to suit an explanation can be illustrated with personal information a user will put in their account. A user may enter a career into their account and a profession, and these attributions to a user will be saved in the XML file. The web page loading certain account information will then access an XML file and use the predefined tags to obtain accurate information.

Test Strategy and Plan

Majority of testing will be done in modularity. The largest divisions in testing will be functional and non-functional, both applied at specific times during development. User Interface will initially be tested on all client end forms and displays for browser compatibility. As functionality is added to associate the client end website with the server, there will be mass amounts of functional testing to assure that each step of development proceeds with fully functional systems. Both functional and non-functional tests will occur dynamically. As more code is added, old code will be tested with the inclusion of new functionality.

The functional end of the system will be tested with a white box approach. Since the team of developers is actually a single person, there is no possibility to do any external form of testing. Integrated testing will be used as software will be appended to iteratively and the goal is to maintain a working system at any given point during development. The following test schedule will adhere to the various points in development to assure each milestone is addressed with a specific and efficient series of tests.

Test Schedule

Approaching Milestone	Test details
1	Test web tree to assure all protected areas are protected, Create SQL queries with test accounts
2	Test account creation UI, Test account creation System, Test login/logout UI, Test login/logout System, Test complaint UI, Test complaint system Test search engine UI, Test search engine, Run full modular tests against database
3	Test Rating Widget, Test Rating System, Test Account UI, Test Profile Edit UI Test Profile Edit, Test full account and edit system against database
4	Test data log UI, Test data log system, Test Accounts Management UI, Test Accounts Management System, Test Event Management UI, Test Event Management
5	Test UI configuration, Test JQuery scripts against all pages Test full website under updated UI system

Changes

Overview

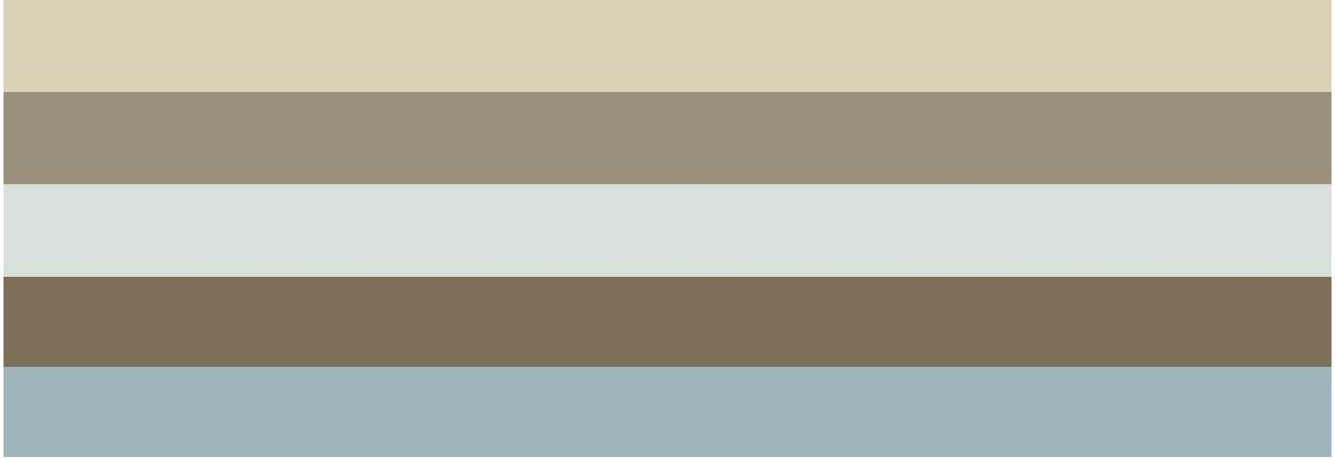
From the initial development planning stages to actual implementation, many obstacles have been discovered as well as areas that can be optimized. Many technical changes have been implemented or have a planned change in implementation as development continues. The host operating system, database, page design, and hosting have been addressed. As the website personality builds and components of the website are fit into the testing server system, aesthetics are also being addressed with detail causing massive changes to development. Issues either causing major stops in development or requiring work arounds have also been addressed.

Changes Specifications

The operating system that was initially intended for development and hosting, as is stated previously, was linux. The LAMP system was to be used. The operating system now is windows, and the only reason for this implementation was the experience that staff have with the current software allowing easier configuration to get towards core development faster. The database now includes tables addressing ratings. Users will now be tracked on ratings and and these ratings will play a role in the buddy system, to allow a user to find further involvement with individuals who they have previously rated highly or not had an experience with at all. Hosting is being done through a dynamic server hosted from a home computer for testing purposes. The domain is called peoplewithpurpose.mooo.com and it is attached to a dynamic DNS software to allow consistent connection. JQuery is playing a large role in situations dealing with dynamic form validation to allow a site involved heavily with registration and authentication system. There are PHP validation scripts to maintain consistency in registration and authentication security, incase javascript is disabled.

Major Issues

The first and foremost issue currently is the dynamic DNS system that has been implemented. Although the initial implementation was smooth and allowed no reason for alarm, bar a few IP reassignment update issues, there are major black out problems with the current domain. A transfer to CDF systems is being considered after submission of phase three assignments with the same domain name. Another issue that was neglected was the lack of personnel. Man power is not currently the issue, the over establishment of functionality within the current website is the major issue. A reassessment of priorities and tasks has yielded a simpler website with implementations of functional necessities rather than eloquent designs. Users can still expect a pleasant experience but their may be obvious lacking sections of the website that may not cripple site functionality but will definitely not be the most polished display of development.



*Pallet adopted from http://www.colourlovers.com/palette/77121/Good_Friends

Figure 1