dnwTennis

A camp registration system

Specifications – Version 2015.1

**Primary Goals – V2015.01**

dnwTennis takes a simplistic approach to managing registrations for a tennis sports camp. V2.1 will be a complete rewrite using Google’s GO language and he MongoDB database. In addition there will be a few high value features added to the package.

Table of Contents

Features Summary 1

ReWrite 2

System tracks more than one camp 2

Logins 2

# Features Summary

* Rewrite Application
* System tracks more than one camp
* Optional Logins
* Email camp participants
  + Allow sender to select one or more participants and shortcuts for all in section and all in camp
* Optional email entry for each camper
* Log changes to DB to provide
  + Database Change history
  + History of single user – Including past camps attended
* Keep historical Camp Data
* Allow > 1 camp to be managed
* Camp Close out – Flag a camp as inactive
* User Registration screens
  + Users can modify their registrations
  + Confirmation email needs to be prettied up
* Management Screens
  + Create a new registration
  + Remember which camp and camper the user was viewing and return there after making modifications
  + Move Camper - Split a registration by moving a camper to a different section
  + Add the timestamp date to the display
  + Allow a manager to review past camps
  + Search for a name – General Search option
  + Display problem with screens where radio buttons scroll in table cell on ipad
* Pro’s Functions
  + Add a piece where the pro can create subsections for each section and assign people in the camp to the subsection
  + Once the pro has assigned someone to a section there should be a notification when a camper or registrations is modified in any way
* Reports
  + Year-end camp summary – This is for charging residents as well as for the pro to see who is coming
    - Count of guests
    - Count of residents

# ReWrite

I have tried to come up with some good reasons for the rewrite but they are all made up. The real reason I am rewriting this is because I want to write a complete application in GO using mongoDB. So there – I said it. The original selection of PHP had more to do with what was supported at Dreamhost where the site is hosted.

The following packages and libraries will be used in the rewrite:

* MGO – Labix.Org
* Gorilla MUX –
* Bootstrap – The Twitter CSS Package

# System tracks more than one camp

* Database changes to handle more than one camp and linkage so we know which camp belongs to which user

# Logins

Logins can be an inhibitor to using the system and introduce a support headache helping people find their logins. It is also silly to ask someone to carry around a new login for a camp registration site. For these reasons the system will support optional login but will not offer a homegrown login, rather it will use OAuth2 logins from the following providers:

* Google
* FaceBook
* Microsoft
* Add Others here they are deemed useful

Users who do not login will be able to register for a camp only. They will not be able to View/Modify or Delete their registrations. Users who do login will be able to perform CRUD operations on registrations associated with the login.

Managers and Monitors will be required to login using OAuth2 credentials.

User rights will be established to allow for appropriate access to the site.

No Login – register for camps

Regular users – Can Add, Modify or Delete their own registrations

Admin Users – Can access the management interface – There are two types of admin users

Monitors – Can access the management page in read only mode

Admins – Can access the management page in r/w mode

TOCONSIDER – If a user does not login and creates a registration then later logs in because they hear they can make changes if they do, will the SW be able to tell which registration(s) belong to the user?

PREVIOUS WRITEUP

The system will be enhanced to allow for more than one camp and to maintain historical data about those camps. If a user has logged in and has an account, they will be able to create a camp that others will be able to register for. This will require logic to have a create option appear on the screen for the user and a set of screens to allow that user to fill in details for the new camp as well as assign rights to control and view camp details to other users.

Create a Camp

Modify an Existing Camp

Cancel a Camp

Delete a camp

ISSUES TO CONSIDER – Do we notify camp participants when a camp is deleted or canceled?

CAMP STATE

Camps can be in one of multiple states

Active – Camp is setup and is taking registrations

Suspended – Camp is setup but not taking registrations

Inactive – Camp is past the registration deadline

THINK HERE – For the DNW camps the registration pages change as the registration deadline comes and goes but participants are allowed to continue signing up with a required confirmation by the camp coordinator. I wonder if for the dnw camps this concept is making things needlessly complicated – The lottery as it is called is really a way to stage registrations up until a certain date after which the camp coordinator will adjust who is in and who is not based on some criteria. The system could handle this for the coordinator with a simple set of rules:

* Camp registration limit – With coordinator override
* After limit is reached new entries go on the waitlist – Case where camp has (n) spaces, Group (a) has (n+1) people and group (b) has (n) Group (a) is first so should get in the camp but group (b) fits better.
* Groups are kept in tact

The current structure for participants is that there is a group leader for a camp who has signed up one or more camp participants. The group leader is also presumed to be responsible for paying for the all the participants.

THINK HERE – There is some benefit in having some information about the individual campers such as email or phone so that camp emails can optionally be sent to campers. If you are a group manager for an all-adult group that could be a benefit but if you are a group manager for a bunch of kids then that is less useful. This implies that to generalize there needs to be some optional information about campers you can enter and an option to sign an individual camper up for camp emails. This can add complexity to the system that should be avoided. Right now it is very simple and needs to stay that way.

Allow a team leader to name the team

**New Features for Camp Managers**

Email all or some of the participants in a camp / Create and email list from the participants list

**Specific Work Items**

Description Entered Completed

**General**

|  |  |  |
| --- | --- | --- |
| Login via OAuth2 services including: Google; Facebook, Microsoft, LinkedIn | 6/1/14 |  |
| Design ACL Function | 6/7/14 |  |
|  |  |  |

**Recode**

|  |  |  |
| --- | --- | --- |
| Recode the entire project using GO | 6/1/14 |  |
| Port Database to from MySql 🡪 MongoDB | 6/1/14 |  |

**Generalization**

|  |  |  |
| --- | --- | --- |
|  | 6/1/14 |  |
|  | 6/1/14 |  |

**Registration Form**