# CMGC Web Site

## PHP Setup

Change the default number of maximum input vars from 1000 to 2000. This allows POSTs to have up to 2000 variables.

In .php/<version>/phprc, add the following line: max\_input\_vars = 2000

After you change this file, you have to SSH into the server and kill the PHP process so the new variable is read (or it will exit on its own if you wait). killall -9 php56.cgi (for PHP version 5.6)

## Occasional Maintenance

1. Remove old log files in “logs” subfolder. There is a log file for each tournament and ipn.log for paypal exchanges. If there are no errors to diagnose, just delete the log files.

## Backups

### Backup WordPress

1. Log in to WordPress (In chrome and MS Edge, I have to log in twice. The first time gives me an error about cookies. The second time works.)
   1. coronadomensgolf.org/wp/wp-login.php
   2. account: cmgcadmin
2. Click on Duplicator
3. Click on Create New
4. Update name field with today’s date
5. Click on Next
6. Click on Build
7. Click on Download Archive and Installer to save locally

### Backup Golf Database

1. Log in to phpMyAdmin (use MS Edge browser)
   1. mysql-1.coronadomensgolf.org
   2. account: cmgcdbadmin
2. Select database coronadomensgolf\_main
3. Click on Export tab
4. Click on Go (format sql) which makes the export and downloads

## Backup PHP Scripts

1. Create compressed folder of C:\Users\albitz\PMA\git\cmgc\Web
2. Copy to backup folder

Backups are in your download folder. Copy to a backup area.

## Setup New Website from Backups

### Setup Golf Database

1. Log onto Dreamhost
2. Go to “Billing & Account” “Manage Account”
3. Scroll to “Databases” and click on “Add a Database”
4. Fill in “Database Name” (and users if needed) and click on “Add new database now!”
5. Login into MySql (example: cmgcdb.paulalbitz.com)
6. Select new database
7. Click Import tab
8. Browse to backup sql file to import and click “Go”

## Setup WordPress

(If something goes wrong after the setup, look in Dreamhost log files using ftp. e.g.: paualb5/logs/paulalbitz.com/https/error.log)

1. create archive with Duplicator plug-in
2. copy installer.php and zip to target folder
3. Run installer.php
4. fill in database host (mysql.paulalbitz.com) WP database (paulalbitz\_com) user (paulalbitzcom) and password
5. install
6. login using installed database admin and password
7. remove installer
8. Copy all php scripts to V2 subfolder
9. Copy the following files to top level of domain: .htaccess, login.php, index.php, paypal\_ipn.php, paypal\_dues\_iph.php, robots.txt
10. Copy apple icons to top level of domain
11. Update login.php with new site information if necessary
12. In WordPress, update paths (coronadomensgolf.org to paulalbitz.com) using Search and Replace tool on all databases. This will update all the menu and media paths.

## PayPal Create Button

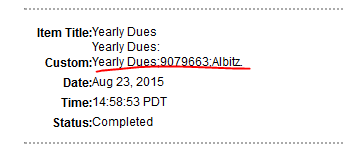
Login

Tools

PayPal Buttons

## Yearly Dues Payment

After paying by PayPal, PayPal will send back a notification to paypal\_dues\_ipn.php at the web site’s top level. The notification will contain the following custom field: “Yearly Dues;9079663;Albitz” The GHIN number is used to find the entry in the Dues table. This data is also displayed in the PayPal transaction:



The Dues table has the following fields:

|  |  |  |
| --- | --- | --- |
| Field Name | Type | Value |
| Year | smallint(6) | Year for which dues apply |
| GHIN | int(11) | Player’s GHIN |
| Name | varchar(50) | Players Name |
| Payment | float | Payment reported by PayPal |
| PaymentDateTime | datetime | Payment date and time reported by PayPal |
| Payer Name | varchar(50) | Payer name reported by PayPal |
| PayerEmail | varchar(75) | Payer email reported by PayPal |
| RIGS | tinyint(1) NOT NULL DEFAULT '0' | Resigned in good standing. (Not used) |

The transactions reported from PayPal are recorded under the top level logs folder in a file called dues.<year>.log. The log entries are as follows:

[2015-08-22 15:03 America/Los\_Angeles] GHIN = 9079663, name = Albitz, payment = 3.00

[2015-08-22 15:03 America/Los\_Angeles] Updated player Albitz, Paul payment to 3.00

## Developer Site Activities

Here are changes to make to the database on the development machine. Make sure all email goes to you.

Set all email to palbitz: UPDATE `Roster` set Email = ‘[palbitz@san.rr.com](mailto:palbitz@san.rr.com)’

Here’s how to clean out signups:

Delete table rows: DELETE FROM `SignUpsPlayers` WHERE TournamentKey=30

## Submitting Results

### Scores

The scores results are submitted as an array of key-value pairs named like this: ResultsScores[{0}][Flight]. The values are stored in this object on the web site and eventually written to the SQL database.

**class** Scores {

**public** $TournamentKey;

**public** $Name1;

**public** $GHIN1;

**public** $Name2;

**public** $GHIN2;

**public** $Name3;

**public** $GHIN3;

**public** $Name4;

**public** $GHIN4;

**public** $ScoreRound1;

**public** $ScoreRound2;

**public** $ScoreTotal;

**public** $Flight;

**public** $TeamNumber;

**public** $Date;

}

If the tournament is match play, the results are submitted as an array of key-value pairs named like this: MatchPlayResultsScores[{0}][Round]. The values are stored in this object on the web site and eventually written to the SQL database.

**class** Match {

**public** $TournamentKey;

**public** $Round;

**public** $MatchNumber;

**public** $Name1;

**public** $Name2;

}

### Chits

The chits results are submitted as an array of key-value pairs named like this: ResultsChits[{0}][Flight]. The values are stored in this object on the web site and eventually written to the SQL database.

**class** Chits {

**public** $TournamentKey;

**public** $Name;

**public** $GHIN;

**public** $Score;

**public** $Winnings;

**public** $Flight;

**public** $Place;

**public** $TeamNumber;

**public** $Date;

**public** $FlightName;

}

### Pool

The pool results are submitted as an array of key-value pairs named like this: ResultsPool[{0}][Flight]. The values are stored in this object on the web site and eventually written to the SQL database.

**class** Pool {

**public** $TournamentKey;

**public** $Name;

**public** $GHIN;

**public** $Score;

**public** $Winnings;

**public** $Flight;

**public** $Place;

**public** $TeamNumber;

**public** $Date;

**public** $Hole;

}

In WebAdmin, the pool results can be adjusted. When adjustments are made, a new csv file is written out. This file contains all the key-value pairs that are ultimately sent to the web site. This allows for filtering out any adjusted payouts of $0.

The adjusted file is named with the extension “.adjusted” and the contents look like this:

ResultsPool[0][Winnings],"50"

ResultsPool[0][Flight],"1"

ResultsPool[0][Date],"2015-09-26"

ResultsPool[0][Place],"1"

### Closest To The Pin

**class** ClosestToThePin{

**public** $TournamentKey;

**public** $Date;

**public** $GHIN;

**public** $Name;

**public** $Hole;

**public** $Distance;

**public** $Prize;

**public** $Business;

}

## Tournaments

### Getting Tournament Names

The actual tournament details became so large, that a shorter tournament names object was created. This object has only the items from the tournament that are needed to make a selection.

class TournamentName {

    public $Name;

    public $StartDate;

    public $EndDate;

    public $SignupStartDate;

    public $TournamentKey;

    public $IsEclectic;

    public $MatchPlay;

    public $TeamSize;

    public $IsStableford;

public $AnnouncementOnly;

}