

Server-side PHP code:

In web server root:

ddbootstrap.php

– both builder and app call this single file, which then loads the proper routine from “services” folder. This allows for added security, as the “services” folder can have its permissions reduced to server access only.

In folder “services”:

AdminDeleteCharacter.php

AdminGetCharacters.php

AdminPutCharacter.php

– used by DDBuilder only

common.php

– log code used in all files

db_include.php

– database login details (name, password, etc.). This includes same username/passoword info as WP, as tables reside in WP database. So if WP access info is changed, this file must be updated to reflect the changes.

DDLogin.php –

- this is called by the game, with the accesskey passed in from URL. This takes the place of a standard username/password style login. The routine looks up the user associated with the accesskey and returns the number of dunks remaining for the account, and the list of character IDs that this account is authorized to dunk. Special ID of “all” means the entire list of characters are available.

GetCharacters.php

- this returns is used by game, and takes in a list of IDs, and returns the info for each character (name, head, body, dunk count).

GetDunkList.php-

- this is used by the Stats/Ranking app to get list of characters/stats.

paypal_ipn.php

- this the call-back routine which PayPal calls after a purchase. This routine sends the email to the user, and also creates the database record that has the accesskey, dunks, and list of IDs that the purchase includes. This file needs to be updated when new buy now buttons are created in PayPal. The item ID (not name), is referenced from PayPal to set the dunk count and IDs of characters which the purchase includes.

UpdateDunk.php

- called by the game, this routine increases the dunk count of a character, and also decreases the remaining dunks available for a given user account. If there is no user account (person is playing the free game), or if it's the special unlimited account link, then no character dunk counts are increased.

Database:

Tables are contained within the WP database (db_include.php must be updated if WP passwords are changed).

ddchars

- list of characters and dunk count. Reference this table to obtain character IDs (column: cid).

ddorders

- user account info.. accesskey, dunks remaining, character list, and purchase info.

logs

- purchases are logged here, as well as any errors that occur.

Client-side SWF Flash files:

All .swf files are located in web server root:

DDBuilder.swf

- this the character builder/setup app. Once all characters have been created, this swf should be removed from the server, as anybody who finds this link will be able to add/delete characters from the game.

DonnieDunko.swf -

- the game. Any Client calling this without an accesskey in URL, will receive the free version only (access only to Donnie Dunko character).

DunkRanking.swf

- used by Stats/News page. Accessible by any client.

Adding New Characters Heads to Flash Source

To add new heads to the game, two files must be modified: DonnieDunko.fla & DDBuilder.fla.

Within DonnieDunko.fla, folder “__Main” the movieclip “All_Heads” contains a single head on each frame. Simply add a new frame and import PNG of head into that frame, with 0,0 point being just above neck centered. Do not change order of other existing heads in time-line, as this will mess up all existing characters.

Within DDBuilder.fla, folder “__Main” movieclip “All_Heads”, import new heads into each frame in the same order is the game file (very important), however 0,0 must be in upper left corner of image.

Creating New “Buy Now” Buttons for Pack Purchase

While in PayPal, go to “Merchant Services”, then “My saved buttons” near right side. Select action drop-down next to “WA State Politicians Pack” button, “Create similar button”. Fill in “Item Name” and (very important) fill in Item ID. Make sure you use an ID that you haven't used before. All other options can be kept the same (or modified as you see fit). Click “Create Button” when finished and PayPal will provide you with the button code for website. From this code extract the `hosted_button_id` value. Use the below code, but replace with `hosted_button_id` value from PayPal button:

```
<form action="https://www.paypal.com/cgi-bin/webscr" method="post"><input type="hidden"
name="cmd" value="_s-xclick" />
<input type="hidden" name="hosted_button_id" value="[value from PayPal goes here]" />
<input type="image" name="submit"
src="https://www.paypalobjects.com/en_US/i/btn/btn_buynowCC_LG.gif" alt="PayPal - The
safer, easier way to pay online!" />
<input type="hidden" name="custom" id="custom" value="<?php echo $user_id; ?>">
</form>
```

Use the updated above code within your web page. Note: coping all the button source directly from PayPal will not work, as the above source has the additional line of: `<input type="hidden" name="custom" id="custom" value="<?php echo $user_id; ?>">` which is very important. Also, each page which contains a Buy Button using this code must begin with: `<?php $user_id = get_current_user_id(); ?>`

Make sure you have the “Item ID” as this exact ID will be referenced in the IPN file that PayPal calls after a transaction (`paypal_ipn.php`).

Open `paypal_ipn.php` in editor and find the switch statement at around line 92:

```
switch( $item_number ) {
```

A new case will need to be added to this switch statement for each new “Buy Now” button that you've created. The case statement for the first pack reads:

```
case "Presidential":
```

```
    $charlist = '72,89,101,106';
```

```
    $numberofdunks = 999999;
```

```
    $packpurchase = "true";
```

```
break;
```

So, for example, if you created a new button and you want it to only have characters 17,18 and the new ID was “Assortment_Pack”, you'd add these lines:

```
case "Assortment_Pack":
```

```
    $charlist = '17,18';
```

```
    $numberofdunks = 999999;
```

```
    $packpurchase = "true";
```

```
break;
```

Note: ID of character can be found in `ddchars` table.