# nwkidsshow.com README

## Website Version 8

# Introduction

Django 1.4, upgraded to 1.5.

GAE

Cloud SQL, Cloud Storage

Users

Website layout

etc

# DB Schema

See also nwkidsshow-model.pdf



Note that nwkidsshow\_show\_retailers and nwkidsshow\_show\_exhibitors is an artifact created by django and not me. Regardless, it seems redundant with the other path from retailer to show. I don’t feel like fixing it though.

# Release Notes

## Version 8

BUG FIX: closed one day too soon cuz of \_\_gt and not \_\_gte; BUG FIX: GAE datetime objects in UTC so registration closed several hours too soon; BUG FIX: Retails should be allowed to register until the show is over; New centered layout.

# DEV Environment

I use **PyCharm** because all of the Django and Google App Engine commands I use are built-in, not to mention Git Hub integration.

I also have **MySQL** installed for my local database (imported from PROD, see below) and use **MySQL Workbench** to manage the local db and run queries.

## manage.py commands

when you create a new database, you need to run “syncdb”. But since then, I have not used many manage.py commands. Others of interest might be: “reset”, but maybe dangerous.

# PROD Environment

Using Google App Engine with storage in Google Cloud SQL. See below for login and dashboard information.

## How to deploy

If you want to bump the website version number, you’ll have to:

1. edit app.yaml and change “version”. I am using just integers right now. Don’t confuse “version” with django version.
2. Use the GAE dashboard Version to make the new version the default (after you push it up).

In Pycharm (with the “upper” nwkidsshow project open) select Tools -> google App Engine -> Upload App Engine app… but the “…” is misleading because it does not open another dialog but just goes and does the push.

# ADMIN Dashboards

1. Google APIs console to watch Storage and Cloud-SQL:  
   <https://code.google.com/apis/console/b/0/?pli=1#project:729708433766:sql:instance:nwkidsshow.com:nwkidsshowdb:instance1> (login as [info@nwkidsshow.com](mailto:info@nwkidsshow.com)) and you should see the page below:  
   
2. Cloud Storage Console is where the manual db exports get put.  
   <https://cloud.google.com/console#/c=cloudstorage&pid=729708433766&gcsh=nwkidsshow-backups/> (login as [info@nwkidsshow.com](mailto:info@nwkidsshow.com)). You can click on the file to download to your local machine and then import into MySQL via Workbench.  
   
3. Google App Engine is where you manage running apps.  
   https://appengine.google.com/  
   login as [damien@macielinski.com](mailto:damien@macielinski.com)  
   and you’ll see two apps (I have not been using UAT yet)  
     
   click on nwkidsshow to see a big dashboard with lots to do, like:  
   - Check Logs  
     
   - change versions, or delete old versions not needed:  
     
   - Application settings is a big one with lots of things to configure, many of which I don’t even understand:
4. Github, I have two repositories, one for the “inner” app:  
   <https://github.com/damienmac/nwkidsshow>  
   and one for the outer website:  
   <https://github.com/damienmac/nwkidsshow2>  
   It is linked to PyCharm and all files updated before push or after successful test.
5. DNS is hosted by enom (<http://www.enomcentral.com> as “lmacielinski”)  
   
6. Google Apps is the main app placeholder for our domains, and here is a reminder on how to get to that (and add domain aliases under Domains, or add this App Engine service to new apps/domains).  
   <http://admin.google.com> and login as either info@nwkidsshow.com or info@cakidsshow.com. Which hopefully takes you to <https://admin.google.com/nwkidsshow.com/AdminHome> or <https://admin.google.com/cakidsshow.com/AdminHome> or

# Data from PROD to DEV

Login to Google APIs Dashboard (see above)

Select “Google Cloud SQL” and the nwkidsshow instance, currently:  
 [nwkidsshow.com:nwkidsshowdb:instance1](https://code.google.com/apis/console/b/0/?pli=1#project:729708433766:sql:instance:nwkidsshow.com:nwkidsshowdb:instance1)

In the Actions menu on upper RHS select Export Data

Fill out the dialog below using the Google Storage bucket name and a date in the filename like this for example:  


Click export and the message tells you where it is putting it: Exporting instance nwkidsshow.com:nwkidsshowdb:instance1 to gs://nwkidsshow-backups/2013-07-05-export.gz

Log in to the cloud console – cloud storage link above and download the exported gz file to your local machine (right-click, save-as)

Copy it to NW Kids Show directory and use 7zip to decompress to a new folder with that name.

Open MySQL Workbench -> Admin -> Data Import -> Import from Disk -> Import from Self-Contained File. (The file has a schema so don’t try and specify a default schema).  


Start Import button (bottom right)

Root password for my local database is empty.

Output should indicate success:

22:06:08 Restoring C:\Users\Damien\Documents\NW Kids Show\2013-07-16-export\2013-07-16-exportRunning: mysql.exe --defaults-extra-file="c:\users\damien\appdata\local\temp\tmpb\_jvnd.cnf" --host=localhost --user=root --port=3306 --default-character-set=utf8 --comments < "C:\\Users\\Damien\\Documents\\NW Kids Show\\2013-07-16-export\\2013-07-16-export"22:06:28 Import of C:\Users\Damien\Documents\NW Kids Show\2013-07-16-export\2013-07-16-export has finished

# Some useful SQL queries

select \* from nwkidsshow\_exhibitor

select \* from auth\_user

select \* from nwkidsshow\_exhibitor e, auth\_user u  
where e.user\_id = u.id and  
must\_change\_password = '1';

# How to add a column in PROD (simple changes)

(From the DJANGO book, chapter 10)

STEP 1: make the change in the models.py of the local dev environment.  
For example, in Registration model

booked\_room = models.BooleanField()

STEP 2: run a manage.py sqlall on the local dev environment and see the new column details as they will be expected in the database:

CREATE TABLE `nwkidsshow\_registration` (  
…  
 `booked\_room` bool NOT NULL,  
…  
}

STEP 3: create an ALTER command to add the column:

ALTER TABLE `nwkidsshow\_registration` ADD COLUMN `booked\_room` bool NOT NULL default false;

STEP 3.5 maybe bump the version number on app.yaml and update the details in nwkidsshow\_db\_history.txt

STEP 4: test it on the local dev environment by executing it on MySQL Workbench and examining the table, the new column, and its values. Try the admin console and make sure it comes up

STEP 5: make all the code changes - just in the local dev environment - DO NOT PUSH TO PROD. Test it good.

STEP 6: make a BACKUP or EXPORT of PROD database (see above).

STEP 7: execute the same ALTER statement on PROD.  


STEP 8: push the code changes to PROD. Test it good.

# ToDo

## California Kids Show

[ ] If you want the browser to keep displaying www.friend1.com all the way without ever showing www.mywebsite.com, you should have friend1.com setup as a Google App account, and add their domain into your application (just like you did for your own domain). This way, all traffic to www.friend1.com (or whatever subdomain they pick) goes to your appengine app. And your App should detect the domain of the http request and response accordingly.  
Copied from [this link](http://stackoverflow.com/questions/15994024/how-does-one-google-app-engine-app-serve-multiple-domains).

[ ] will have overlapping registration periods with NWKids show so be careful in this method where I assume there is only ever ONE show to register for at a time: get\_better\_choices(). Maybe I will know which website they are on and can still use this? How ugly are the choices when there is more than one? Try this and see!

## Online Payment

[ ] read the book, make a plan

[ ] https?

[ ] paypal

[ ] credit card

## Misc

[ ] Show needs a room reservation deadline and put it in the “important dates” section.

[ ] Privacy Policy (see below)

[ ] Terms & Conditions (see below)

[ ] maximum number of Tables that can be rented is 2? Enforce in form.

[ ] finish centering by putting form error messages in a useful place; change span to div?

[ ] really good test plan around all dates & deadlines

[x] add more useful columns to the Registrations admin console

[x] add “booked\_room” Boolean to registration object and add to admin console.

[x] DEBUG off (will I need error pages?)

[x]REPORT on LINEs at a show, make the NAME a link to the contact info for that exhibitor!

[x]Room Number for exhibitos for a show.

[x]Excel export of reports

[x] custom error pages? <https://docs.djangoproject.com/en/dev/topics/http/views/#customizing-error-views>

[ ] Everything is now online including the exhibitor list for the retailers. I would suggest to login and add Katy's name next to In Play Showroom under "Edit my profile". Sorry, we didn't add space for assistant names. This is good info for my webmaster. :)

[ ] configure django in GAE to send email and enable ADMINS among other things.

[ ]After login, show a green success/advising bar at the top of the activities page, or next page (?)

[ ]After create a new user on /add-user/, take her back to that empty screen. Make an admin page with links to admin stuff?

[ ] robots.txt file?

[ ] Comments box from users (needs email first?)

[ ] enforce strong passwords

[ ] searchable directories of exhibitors and retailers?

[ ] what happens when you make a user not ACTIVE in Django? Test it out. Is this how we want to keep people off the system when no longer in NWKS?

## Migration to 1.5

[x] Localflavor changes: <https://docs.djangoproject.com/en/dev/topics/localflavor/#localflavor-how-to-migrate>

1. Go to <https://github.com/django/> and find the package for your country.
2. from django.contrib.localflavor.fr.forms import FRPhoneNumberField  
   ...to this:  
   from django\_localflavor\_fr.forms import FRPhoneNumberField
3. WILL GOOGLE SUPPORT THIS? NO – so I copied it in locally and will try and push it up with my other content.

## FOOTER

[x] Copyright using template for year so don’t have to update in 2014 etc

[x] Logos: Python, Django, GAE

[x] website version #

# Terms & Conditions

## Exhibitors

You agree to book your own room by the “room reservation deadline” and pay for room rental and all fees, taxes, and charges incurred. A link to the Personalized Group Web Page is provided and should be used to reserve your room.

You consent to allow the Hotel to provide Laurel Event Management with information about your room reservation.

Room cancellations within 72 hours or no-shows will incur a Hotel bill of one night’s suite and tax which is your responsibility to pay to the Hotel.

Early departure will incur a $50 fee imposed by the Hotel and payable to the Hotel. Additional fees from Laurel Event Management may apply.

Overnight parking: $5; day parking: free.

Check-in at 3:00; Check-out by noon.

$15 per table, max 2 ($25 if late). Taxes and service charge apply.

Hotel room furniture shall not be removed from the room. Moved furniture must be replaced before departure.

## Retailers

…

# Privacy Policy

This website shares exhibitor contact information (provided by the exhibitor) with registered Northwest Kids Show retailers. This website shares retailer contact information (provided by the retailer) with registered Northwest Kids Show exhibitors. This website does not share your contact or personal information with any other parties.

This privacy policy is subject to change without notice and was last updated on July 01, 2013. If you have any questions feel free to contact me directly here: info@nwkidsshow.com.

Better? <http://www.freeprivacypolicy.com/free-privacy-policy.html>

# Requirements (BDD)

Here are some requirements surrounding dates so we can make a better test plan around dates and not be caught with last minute bugs like in the September 2013 NWKS.

## Story: Exhibitors Are Assessed A Late Fee

**In Order To** encourage exhibitors to register early  
**As An** even planner  
**I want to** automatically assess a "late fee" when exhibitors register after a show's "late date".

#### Scenario 1: When an exhibitor registers on or before a show's late\_date the late\_fee is $0.

**Given** the date an exhibitor registers on-line  
**When** the date is before the show’s late\_date  
**Then** no late\_fee is assessed  
**And** the invoice line-item for late\_fee is displayed as $0

**Given** the date an exhibitor registers on-line  
**When** the date is the same as the show’s late\_date  
**Then** no late\_fee is assessed  
**And** the invoice line-item for late\_fee is displayed as $0

#### Scenario 2: When an exhibitor registers after a show's late\_date the late\_fee is automatically assessed.

**Given** the date an exhibitor registers on-line  
**When** the date is after the show’s late\_date  
**Then** the late\_fee is assessed  
**And** the invoice line-item for late\_fee is displayed as $75 (or $100 after rate hike)

## Story: Exhibitors Can No Longer Register

**In Order To** plan room assignments and show details  
**As An** event planner  
**I want to** stop exhibitors from registering after the “registration closed date”.

#### Scenario 1: Exhibitors can register on or before a show's closed\_date.

**Given** the date an exhibitor registers on-line  
**When** the date is before the show’s closed\_date  
**Then** the exhibitor is allowed to register  
**And** the date of registration is recorded

**Given** the date an exhibitor registers on-line  
**When** the date is the same as the show’s closed\_date  
**Then** the exhibitor is allowed to register  
**And** the date of registration is recorded

#### Scenario 2: Exhibitors cannot register after a show’s closed\_date

**Given** the date an exhibitor registers on-line  
**When** the date is after the show’s closed\_date  
**Then** the exhibitor is not allowed to register

## Story: Retailers Can No Longer Register

**In Order To** allow as many Retailers to attend as possible  
**As An** event planner  
**I want to** allow Retilers to register throughout the show, up to and including the end\_date.

#### Scenario 1: Retailers can register on or before a show's end\_date.