# nwkidsshow.com cakidsshow.com README

## Website Version 15

# Introduction

Django 1.4, upgraded to 1.5.

GAE

Cloud SQL, Cloud Storage

Users

Website layout

Etc

# Release Notes

## Version 15

Re-built my local dev environment after disk crash. Updated show times just for CKS.

## Version 14

Allow Exhibitors to run the reports on other exhibitors. Allow superusers and testex to register in prod (and dev) and pay through the Braintree test account instead of the real one. Use 4111111111111111. Fixed the activity links to buttons and lined-up, re-worded.

## Version 13

Used some app.yaml regex magic and now serve up the apple icons correctly – no more 404s in the logs for these. Note: serving up static files from GAE does not work – not allowed to open and read files in static directories.

## Version 12

Combine the registration form and credit card checkout onto one page. Dynamically update the invoice and total as the registration form is filled out. Display both form and transaction errors back on the same page with entered info re-filled.

Added number of Rooms and bed type to the registration form and invoice (along with number of tables).

Fix admin interface search box (only search on text fields and follow the inheritance); Fix room display width; Show retailer company on registration page; No errors on add-user not finding a banner; Seed new cakidsshow retailers; Seed Laurie and Damien into both exhibitors and retailers.

## Version 11

New banner and colors, fonts to match. New Menu Bar. New favicon for cks. Unique banner per page with a somewhat unique image. Add-user admin page gets a venue picker. Applied the new website to NWKS as well, split the different css out into different files.

## Version 10

Braintree integrated for payment processing (Sandbox and Prod). Don’t let them register for shows already registered for. Don’t let them pay for registrations they have already paid for. Privacy Policy.

## Version 9

cakidsshow.com supportted, tested, deployed, all from same codebase and database. Use app.yaml to serve up different favicons for the two venues.

## 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.

# 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.

# 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>
7. Braintree is our payment gateway. There is a sandbox account (damienmac) at https://sandbox.braintreegateway.com and a production environment (for both nwkidsshow and cakidsshow) at <https://www.braintreegateway.com> (lmacielinski).

# 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.

# Rebuilding the DEV environment

Trying to capture some notes from when my disk died and I had to rebuild things. Everything went smoothly:

Install PyCharm (3.0 this time)

Install MySQL – this was odd, using community edition and then MySQL admin.  
Making Python work with MySQL was hard: find a pre-built installer MySQL-python-1.2.3.win-amd64-py2.7.exe and run it – don’t mess around with pip, easy\_install, etc. I think this installs the connector and MySQLdb or something. There is a1.2.5 out now, maybe update? (<http://www.lfd.uci.edu/~gohlke/pythonlibs/>) (<https://pypi.python.org/pypi/MySQL-python/1.2.5>)

I did not have the following libraries checked in to GitHub (note I did not run any installers for these and did not copy from Lib/site-packages)

braintree – download the zip and copied just the braintree folder into the tree (under top folder).

django\_localflavor\_us – zip from github, copy just that folder into tree (under top folder).

django\_localflavor\_ca - zip from github, copy just that folder into tree (under top folder).

xlwt - zip from github, copy just that folder into tree (under top folder).

The admin console now has no css, you need to “collect” the static assets into C:\Users\Damien\Documents\GitHub\nwkidsshow2\nwkidsshow\static\admin which is determined by the settings.py static settings. Just run the Pycharm django manage command: Tools -> Run manage.py task -> collectstatic and say “yes” and you’re good to go.

# ToDo

[ ] separate NWKS and CKS lines lists (because some reps in CA can’t show those lines in NW) and right now there is one list per exhibitor regardless of show.

[ ] is there a way to figure out duplicate line names for each show?

[ ] I need to see how many assistants are attending.

[ ] Add a “special requests” text box to the form.

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

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

[ ] email from the app? Or Mail Chimp?

[ ] log login attempts that failed?

[x] Exhibitors see the Exhibitors Report.

[x] clean up the activity pages, use buttons?

[x] Allow admins to register and not get charged and keep them hidden form reports.

[x] new cks favicon using teal color?

## California Kids Show

[x] version of app.yaml to push to caks gae. Use same db in cloud. Conditionalize model (“venue”) view (from hostname), controller (from hostname), and settings (from app id) to display one or the other site.

[ ] 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!

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

[x] new banner, buttons, color scheme, fonts, spacing applied.

[x] new exhibitor accounts

[x] new retailer accounts

## Online Payment

[x] read the book, make a plan

[ ] https?

[ ] paypal

[x] credit card – Braintree integration

[x] get caks PROD keys into settings.py and publish

[x] Privacy Policy (see below). Publish and send link to Braintree for review.

[ ] Terms & Conditions (see below)

## Misc

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

[ ] really good test plan around all dates & deadlines

[x] apple icons support (no more 404’s on those requests).

[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>

[x] robots.txt file?

[ ] 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?

[ ] 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>

Login URL: [**http://www.freeprivacypolicy.com/login.htm**](http://www.freeprivacypolicy.com/login.htm)   
Email Address: **damienmac@yahoo.com**   
Password: **LoL8Mx**

**What information do we collect?**   
  
We collect information from you when you register on our site or place an order.   
  
When ordering or registering on our site, as appropriate, you may be asked to enter your: name, e-mail address, mailing address, phone number or credit card information.  
  
**What do we use your information for?**   
  
Any of the information we collect from you may be used in one of the following ways:   
  
; To personalize your experience  
(your information helps us to better respond to your individual needs)  
  
; To process transactions

Your information, whether public or private, will not be sold, exchanged, transferred, or given to any other company for any reason whatsoever, without your consent, other than for the express purpose of delivering the purchased product or service requested.

**How do we protect your information?**   
  
We implement a variety of security measures to maintain the safety of your personal information when you place an order or enter, submit, or access your personal information.   
  
We offer the use of a secure server. All supplied sensitive/credit information is transmitted via Secure Socket Layer (SSL) technology and then encrypted into our Payment gateway providers database only to be accessible by those authorized with special access rights to such systems, and are required to?keep the information confidential.  
  
After a transaction, your private information (credit cards, social security numbers, financials, etc.) will not be stored on our servers.  
  
**Do we use cookies?**   
  
Yes (Cookies are small files that a site or its service provider transfers to your computers hard drive through your Web browser (if you allow) that enables the sites or service providers systems to recognize your browser and capture and remember certain information  
  
We use cookies to understand and save your preferences for future visits.  
  
**Do we disclose any information to outside parties?**   
  
We do not sell, trade, or otherwise transfer to outside parties your personally identifiable information. This does not include trusted third parties who assist us in operating our website, conducting our business, or servicing you, so long as those parties agree to keep this information confidential. We may also release your information when we believe release is appropriate to comply with the law, enforce our site policies, or protect ours or others rights, property, or safety. However, non-personally identifiable visitor information may be provided to other parties for marketing, advertising, or other uses.  
  
**California Online Privacy Protection Act Compliance**  
  
Because we value your privacy we have taken the necessary precautions to be in compliance with the California Online Privacy Protection Act. We therefore will not distribute your personal information to outside parties without your consent.  
  
As part of the California Online Privacy Protection Act, all users of our site may make any changes to their information at anytime by logging into the site and going to the 'Edit My Profile' page.  
  
**Childrens Online Privacy Protection Act Compliance**   
  
We are in compliance with the requirements of COPPA (Childrens Online Privacy Protection Act), we do not collect any information from anyone under 13 years of age. Our website, products and services are all directed to people who are at least 13 years old or older.  
  
**Terms and Conditions**   
  
Please also visit our Terms and Conditions section establishing the use, disclaimers, and limitations of liability governing the use of our website at <http://www.nwkidsshow.com>  
  
**Your Consent**   
  
By using our site, you consent to our [online privacy policy](http://www.freeprivacypolicy.com/).  
  
**Changes to our Privacy Policy**   
  
If we decide to change our privacy policy, we will update the Privacy Policy modification date below.   
  
This policy was last modified on August 29, 2013  
  
**Contacting Us**   
  
If there are any questions regarding this privacy policy you may contact us using the information below.   
  
http://www.nwkidsshow.com  
PO Box 2791  
Tualatin, OR 97062  
US  
info@nwkidsshow.com  
  
This policy is powered by Trust Guard [PCI compliance](http://www.trust-guard.com/PCI-Compliance-s/65.htm).

# 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** event 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 Retailers 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.

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

**Given** the date a retailerregisters on-line  
**When** the date is the same as the show’s end\_date  
**Then** the retailer is allowed to register  
**And** the date of registration is recorded

#### Scenario 2: Retailers cannot register after a show’s end\_date

**Given** the date a retailer registers on-line  
**When** the date is after the show’s end\_date  
**Then** the retailer is not allowed to register

# Test Cases

## NOTES

Daylight Saving Time (United States) 2013 began at 2:00 AM on Sunday, March 10 and ends at 2:00 AM on Sunday, November 3.

Portland = GMT – 7 during DST

Portland = GMT – 8 when not in DST.

But all of my logic only considers the Date not Date & Time (Datetime). So maybe it isn’t important to consider DST at all…

GAE servers are on UTC time with no localized time zone, no daylight savings time.

All of my dates are US-Pacific time zone, also abiding by DST.

So it IS important to know the difference between UTC and Pacific since this is the problem I already had once and want to catch again.

## SETUP

In DEV, Create a show in the future with  
 late\_date = 2013-12-13  
 closed\_date = 2013-12-27  
 start\_date = 2014-02-23  
 end\_date = 2014-02-25

Add some print() to make sure what server is reporting is what we expect, and show our conversion while we’re at it.

CASE 1a: Exhibitor Late Fee before late\_date – (Pacific and UTC match Date)

Change server date to **UTC:2013-12-12 22:00** (it is 12/12 in Pacific)  
Log-in as testex, unregister if previously registered.  
Register for the show  
See that no late fee is assessed

CASE 1b: Exhibitor Late Fee before late\_date – (pacific and UTC differ on Date)

Change server date to **UTC:2013-12-13 03:00** (it is 12/12 in Pacific!)  
Log-in as testex, unregister if previously registered.  
Register for the show  
See that no late fee is assessed

CASE 1c: Exhibitor Late Fee on late\_date – (pacific and UTC match on Date)

Change server date to **UTC:2013-12-13 22:00** (it is 12/13 in Pacific!)  
Log-in as testex, unregister if previously registered.  
Register for the show  
See that no late fee is assessed

CASE 1d: Exhibitor Late Fee on late\_date – (Pacific and UTC differ on Date)

Change server date to **UTC:2013-12-14 03:00** (it is 12/13 in Pacific!)  
Log-in as testex, unregister if previously registered.  
Register for the show  
See that no late fee is assessed

CASE 1e: Exhibitor Late Fee after late\_date – (Pacific and UTC match on Date)

Change server date to **UTC:2013-12-14 22:00** (it is 12/14 in Pacific!)  
Log-in as testex, unregister if previously registered.  
Register for the show  
See that late fee IS assessed

# Other Test Cases

How to make a Test Purchase in PROD: THIS NO LONGER WORKS – THINK OF SOMETHING ELSE

1. Setup: delete testex registration object from a show (if exists)
2. Setup: remove testex from Show exhibitors list (if present)
3. Register for the show, see invoice and call to purchase with credit card.
4. Using the admin console, modify the Registration object total to be $2.0
5. Refresh the invoice page and make sure total is $2
6. Click checkout and pay for it using Laurie’s business cc
7. Visit the Braintree console to see the charge, “submitted” state
8. Later, visit the console again later to make sure it processed
9. Visit the First tech checking account to see the $2 charge and good merchant description.
10. 2 days later, ensure the amount got deposited back into checking: