CDM.DEPAUL.EDU

- 90 full time faculty
- 300 courses/quarter
- 1858 graduate students in 20 programs (CS, IS, ...)
- 1763 undergraduate students in 16 programs
- DHS and NSA center of excellence

MASSIMO DI PIERRO

- PhD in Physics (Lattice Quantum Chromodynamics)
- "Director" MS in Computation Finance
- Interests: Numerical Algorithms, Web Development

WEB DEVELOPMENT WITH WEB2PY

Massimo Di Pierro

School of Computing and Digital Media
DEPAUL UNIVERSITY
Chicago, IL

my job is to make web development easy

my job is to make web development easy

easy != dumbed down
easy != visual programming

easy => more intuitive / less error prone
easy => more expressive
easy => more powerful syntax

easy is not just for kids
easy means experienced developers can concentrate on what is important: algorithms
easy means less development and maintenance costs

Disclaimer: I do not claim any success. I am just trying....

WE2PY: BATTERIES INCLUDED

web server
ssl enabled

DAL + database
auto-migrations SQLite

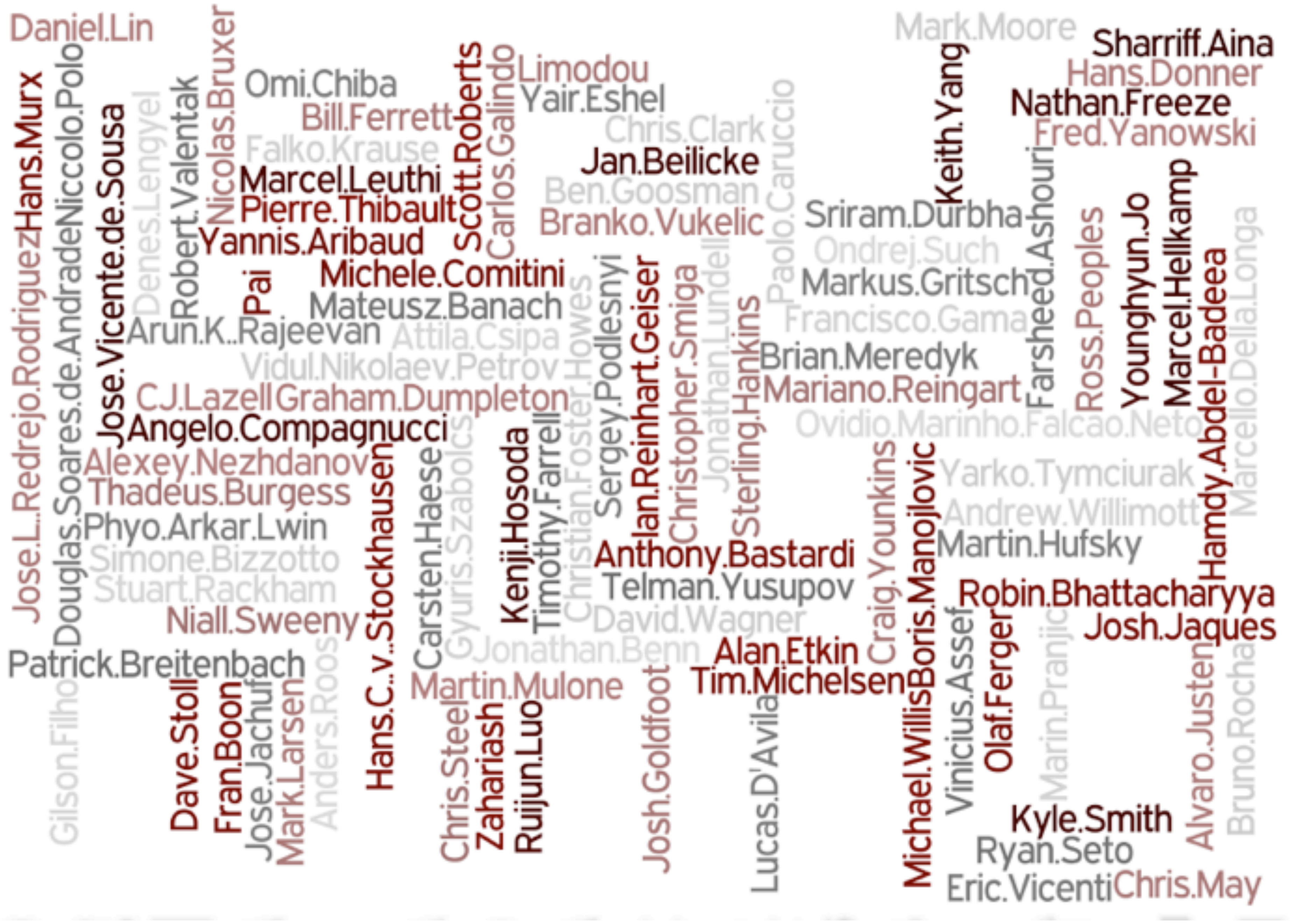
web IDE
design, deploy, manage

html, xml, json, rss, ics, pdf, rtf,
xmlrpc, jsonrpc, soap,
ldap, pam, janrain, dropbox, google,
CAS, OpenID, oauth 1&2, x509
marmin, markdown,
google wallet, authorize.net, stripe.com
memcache, redis
twitter bootstrap

.zip

No installation. No configuration. Just Unzip and Click!

WEB2PY CONTRIBUTORS



2011

2012

Ideas we borrowed

- Model View Controller on WSGI (like everybody else)
- w2p files (like Java's Web application ARchives)
- Routing mechanism (like Django, but optional like Rails)
- Pure Python Template Language (like Mako)
- Helpers (like Rails) but easier: DIV, SPAN, A, ...
- web based database interface (like Django admin)

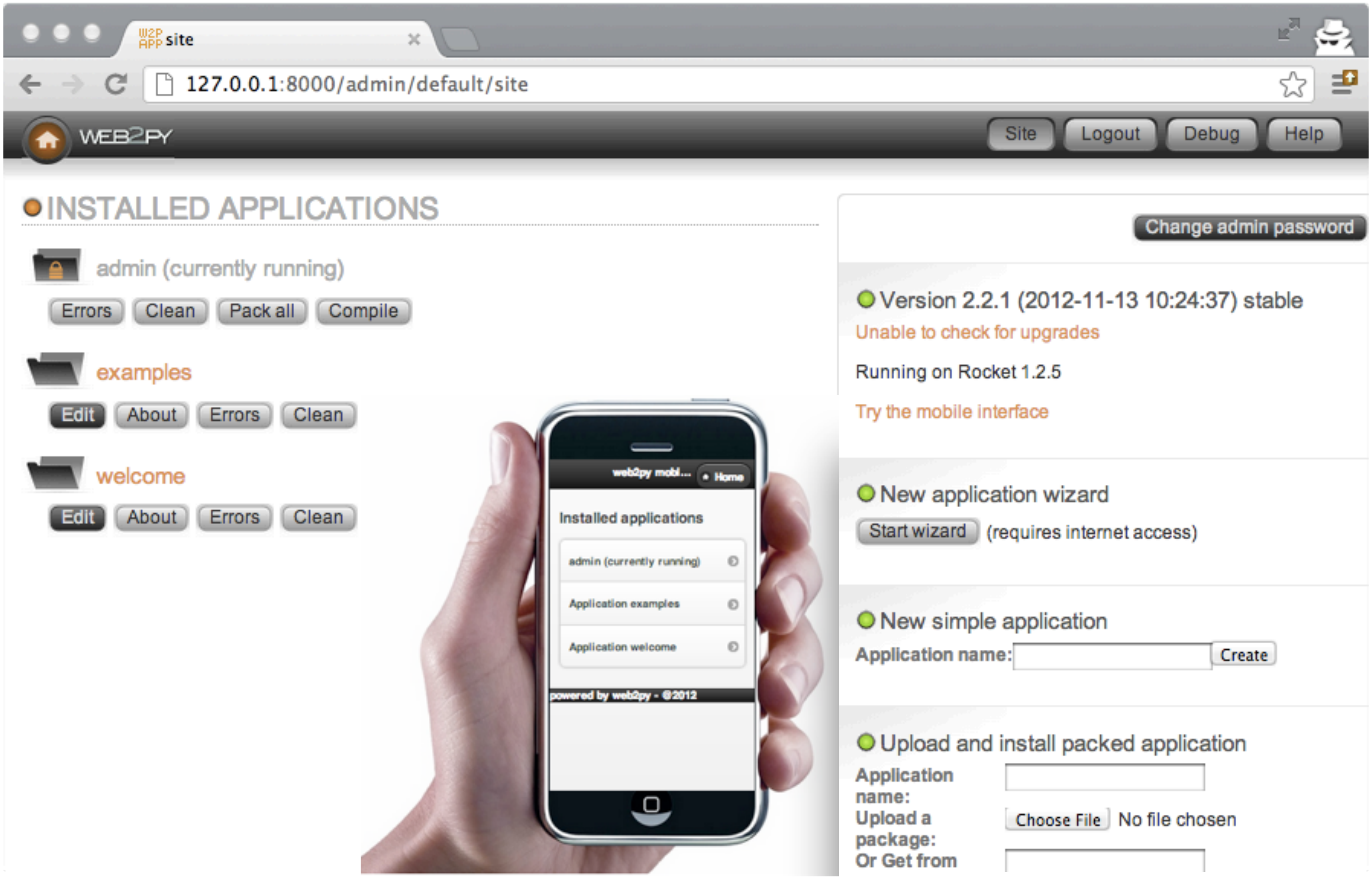
Ideas we had ...

- Always backward compatible (since 2007, 2.5, 2.6, 2.7, pypy, jython)
- One click deploy (Windows and Mac binaries, USB drive)
- No configuration, no dependencies, and secure by default
- Everything has default (DRY)
- Multi project and multi db but share nothing by default
- Web based IDE (development, editor, deployment, management, translations, testing, debugger, version control) shell optional
- Automatic DB migrations (CREATE and ALTER table)
- Plugins / Components / Ajax with Digitally Signed URLs

... Ideas we had

- Role Based Access Control with pluggable authentication modules (openid, dlap, cas, oauth pam, janrain, google, dropbox)
- Every app is a Central Authentication Service consumer and provider.
- Built-in portable cron and master/workers task scheduler
- Full Auditing for all tables
- Ajax embeddable crud & grid controls

Web based IDE "admin" with hot plug and play of multiple apps



INSTALLED APPLICATIONS

admin (currently running)

examples

welcome

Version 2.2.1 (2012-11-13 10:24:37) stable

Unable to check for upgrades

Running on Rocket 1.2.5

Try the mobile interface

New application wizard

Start wizard (requires internet access)

New simple application

Application name:

Upload and install packed application

Application name:

Upload a package: No file chosen

Or Get from

Installed applications

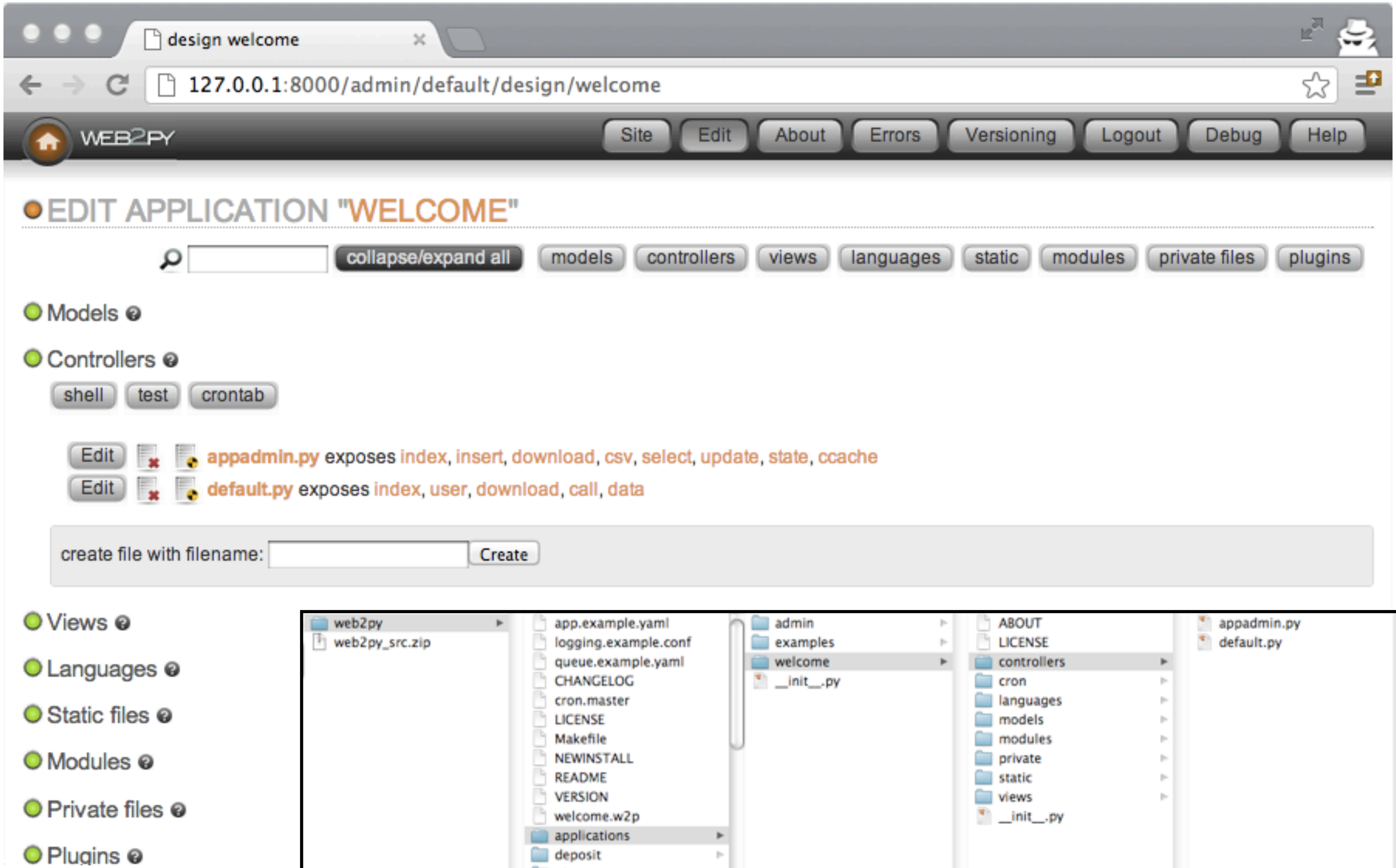
admin (currently running)

Application examples

Application welcome

powered by web2py - @2012

Thin-IDE: only shows file system, no metadata



EDIT APPLICATION "WELCOME"

Models

Controllers

appadmin.py exposes index, insert, download, csv, select, update, state, ccache

default.py exposes index, user, download, call, data

create file with filename:

Views

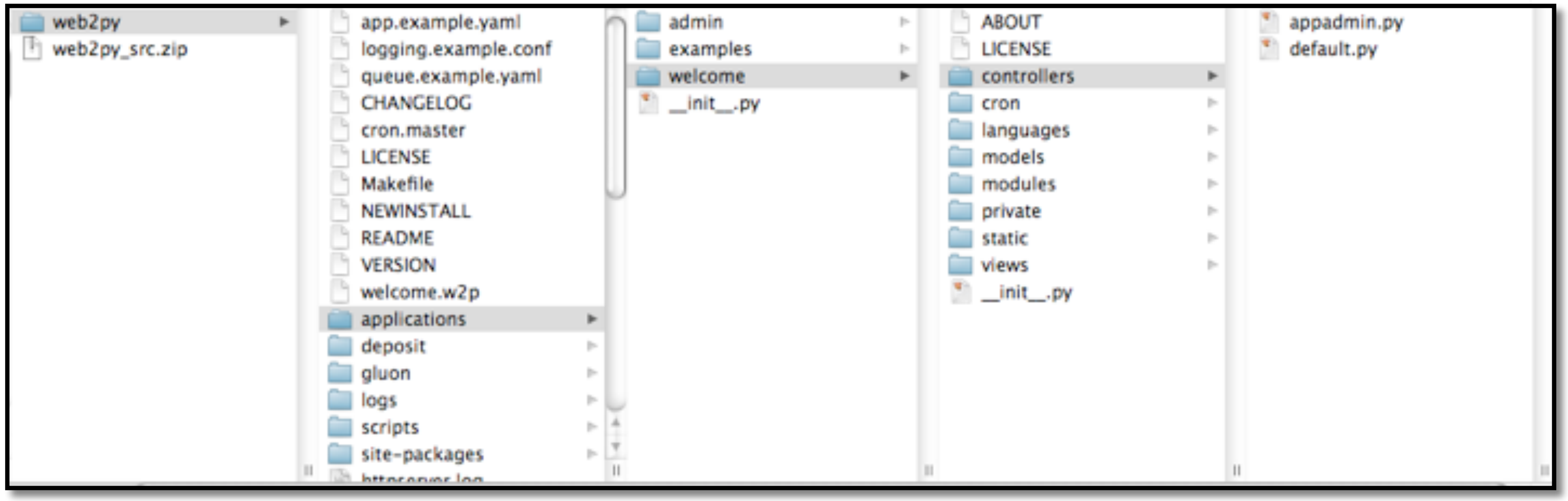
Languages

Static files

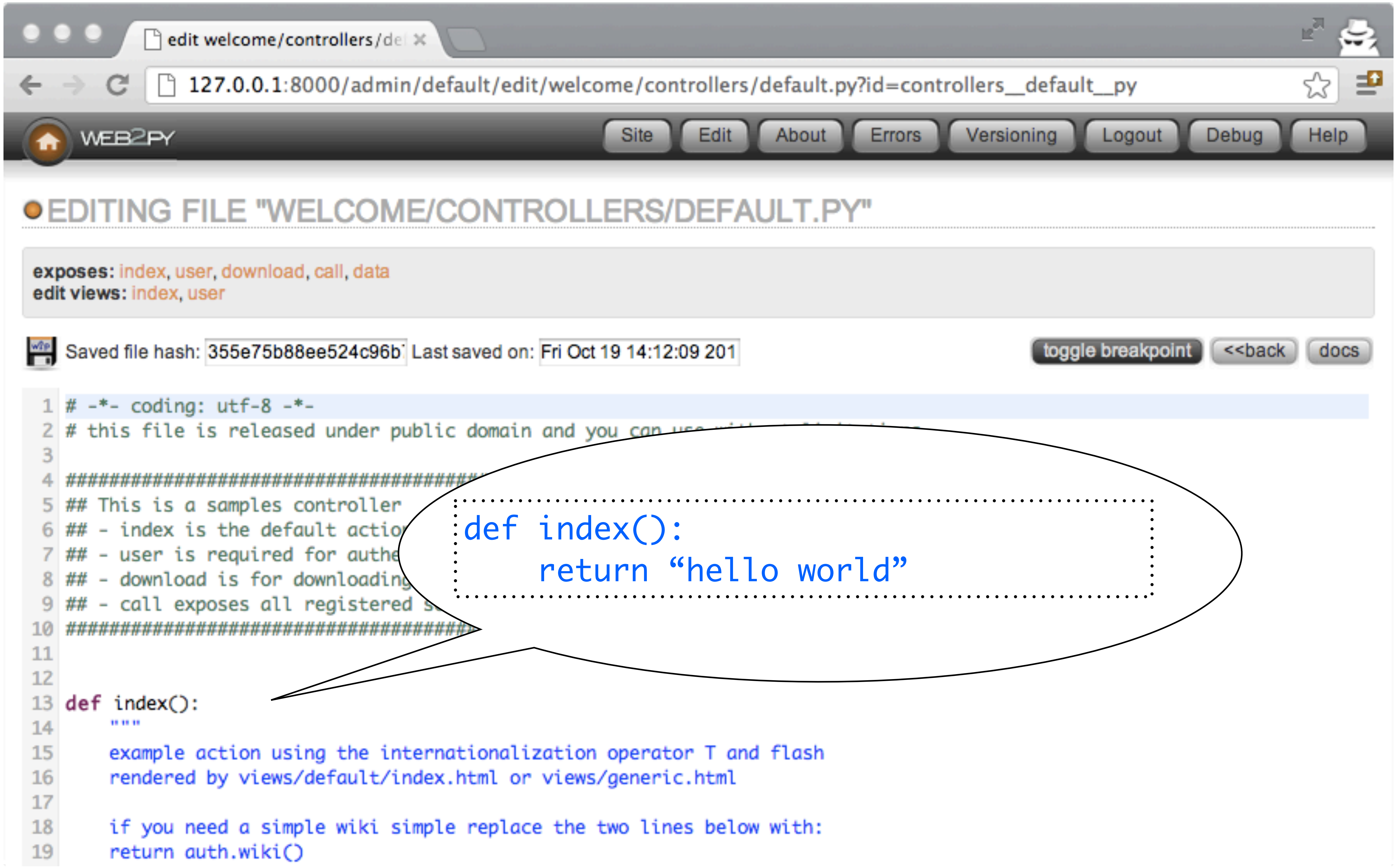
Modules

Private files

Plugins



Web based editor (code-mirror)


edit welcome/controllers/de
127.0.0.1:8000/admin/default/edit/welcome/controllers/default.py?id=controllers__default__py
WEB2PY
Site
Edit
About
Errors
Versioning
Logout
Debug
Help
toggle breakpoint
<<back
docs

EDITING FILE "WELCOME/CONTROLLERS/DEFAULT.PY"

exposes: index, user, download, call, data
edit views: index, user

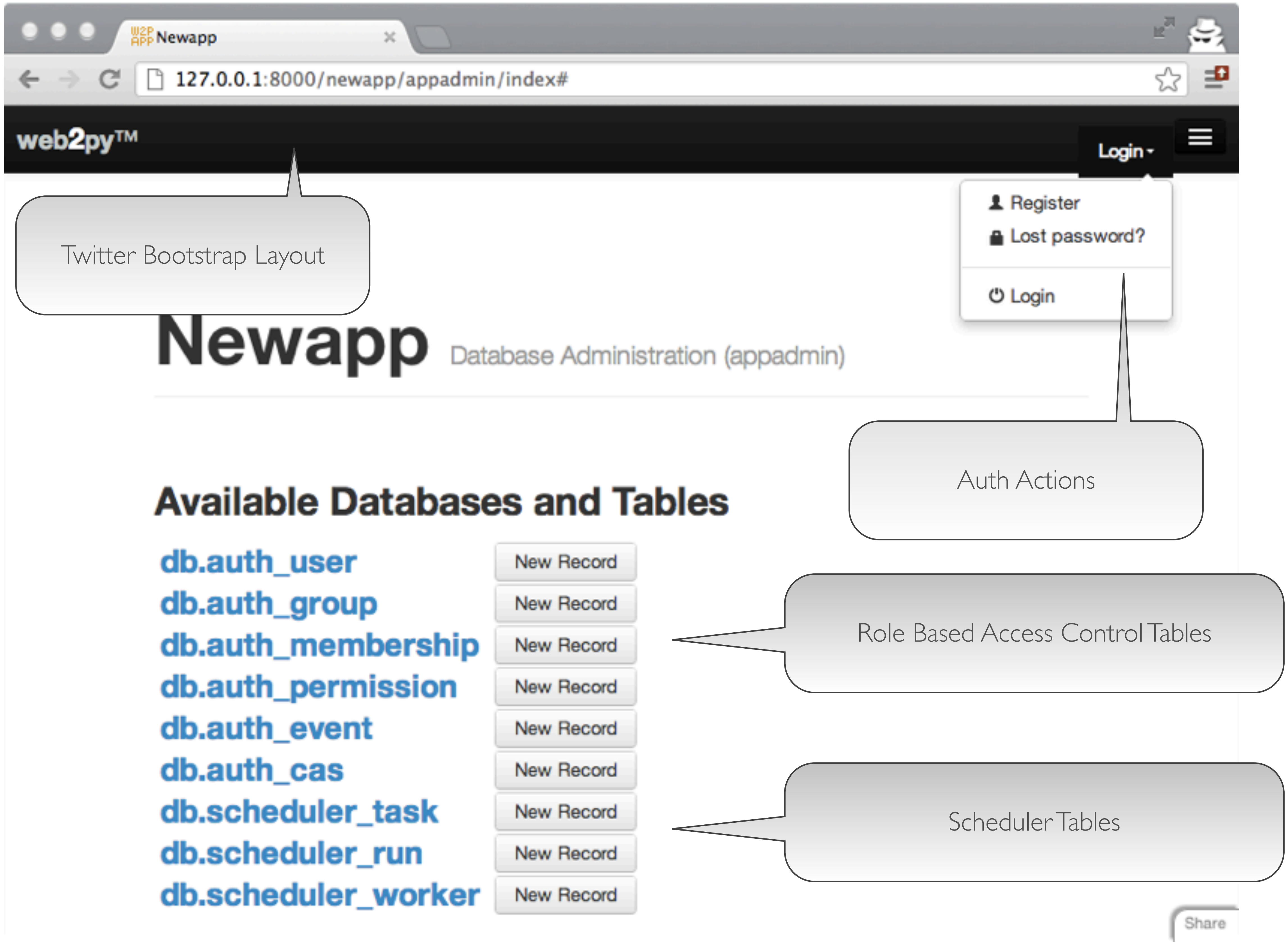
Saved file hash: 355e75b88ee524c96b Last saved on: Fri Oct 19 14:12:09 201

```
# -*- coding: utf-8 -*-
# this file is released under public domain and you can use wi

########################################
## This is a samples controller
## - index is the default actio
## - user is required for authe
## - download is for downloading
## - call exposes all registered s
########################################


def index():
    """
    example action using the internationalization operator T and flash
    rendered by views/default/index.html or views/generic.html

    if you need a simple wiki simple replace the two lines below with:
    return auth.wiki()
```

```
def index():
    return “hello world”
```

Web based database administration (per app)

SQLite, MySQL, PotsgreSQL, MSSQL, Firebid, Oracle, DB2, Ingres, Informix, Ingres, Sybase, GAE, ...

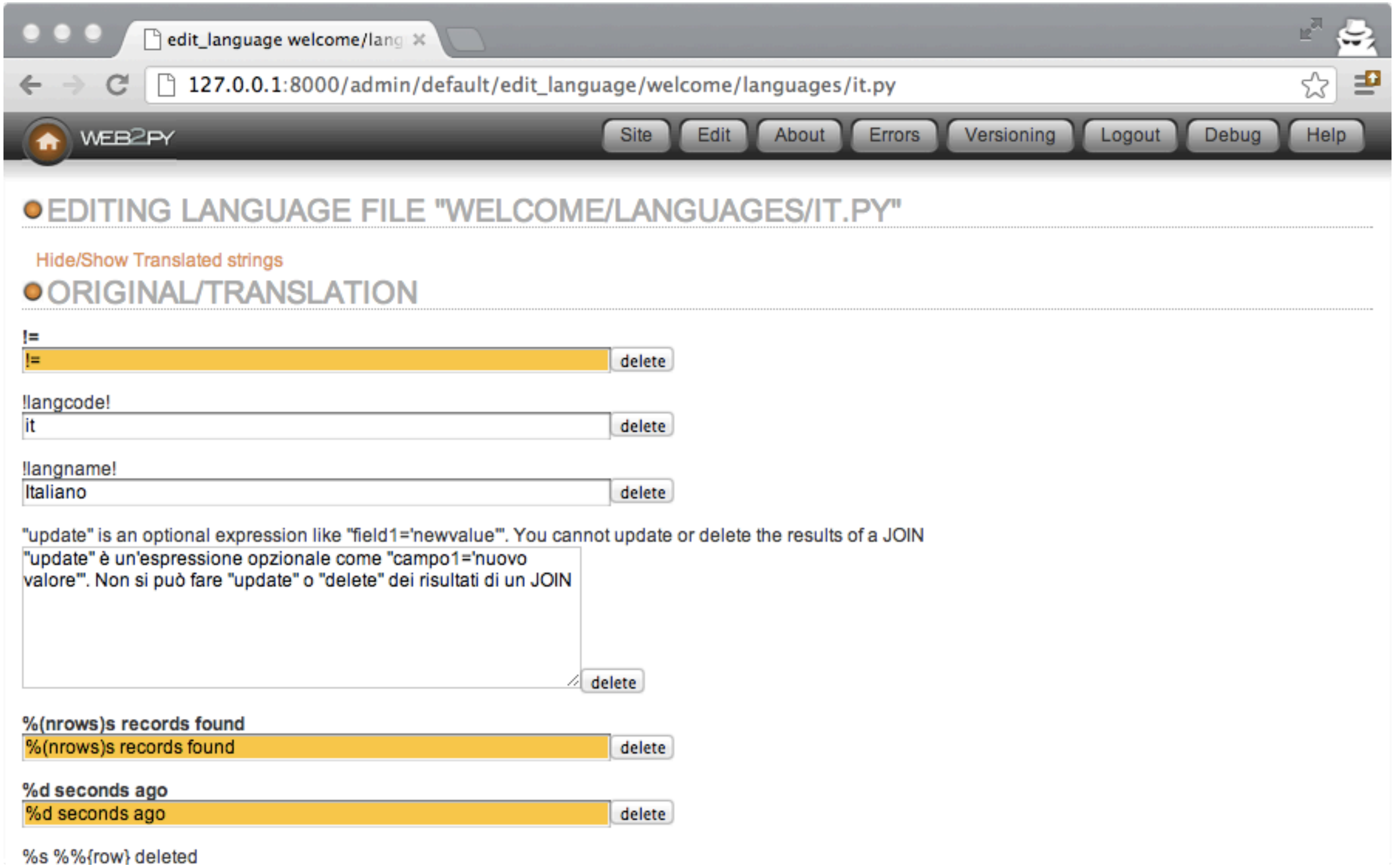


Newapp Database Administration (appadmin)

Available Databases and Tables

db.auth_user	New Record
db.auth_group	New Record
db.auth_membership	New Record
db.auth_permission	New Record
db.auth_event	New Record
db.auth_cas	New Record
db.scheduler_task	New Record
db.scheduler_run	New Record
db.scheduler_worker	New Record

Web translation page for internationalization (per app)


edit_language welcome/lang
127.0.0.1:8000/admin/default/edit_language/welcome/languages/it.py
WEB2PY
Site
Edit
About
Errors
Versioning
Logout
Debug
Help

EDITING LANGUAGE FILE "WELCOME/LANGUAGES/IT.PY"

Hide/Show Translated strings

ORIGINAL/TRANSLATION

!=
!= delete

!langcode!
it delete

!langname!
Italiano delete

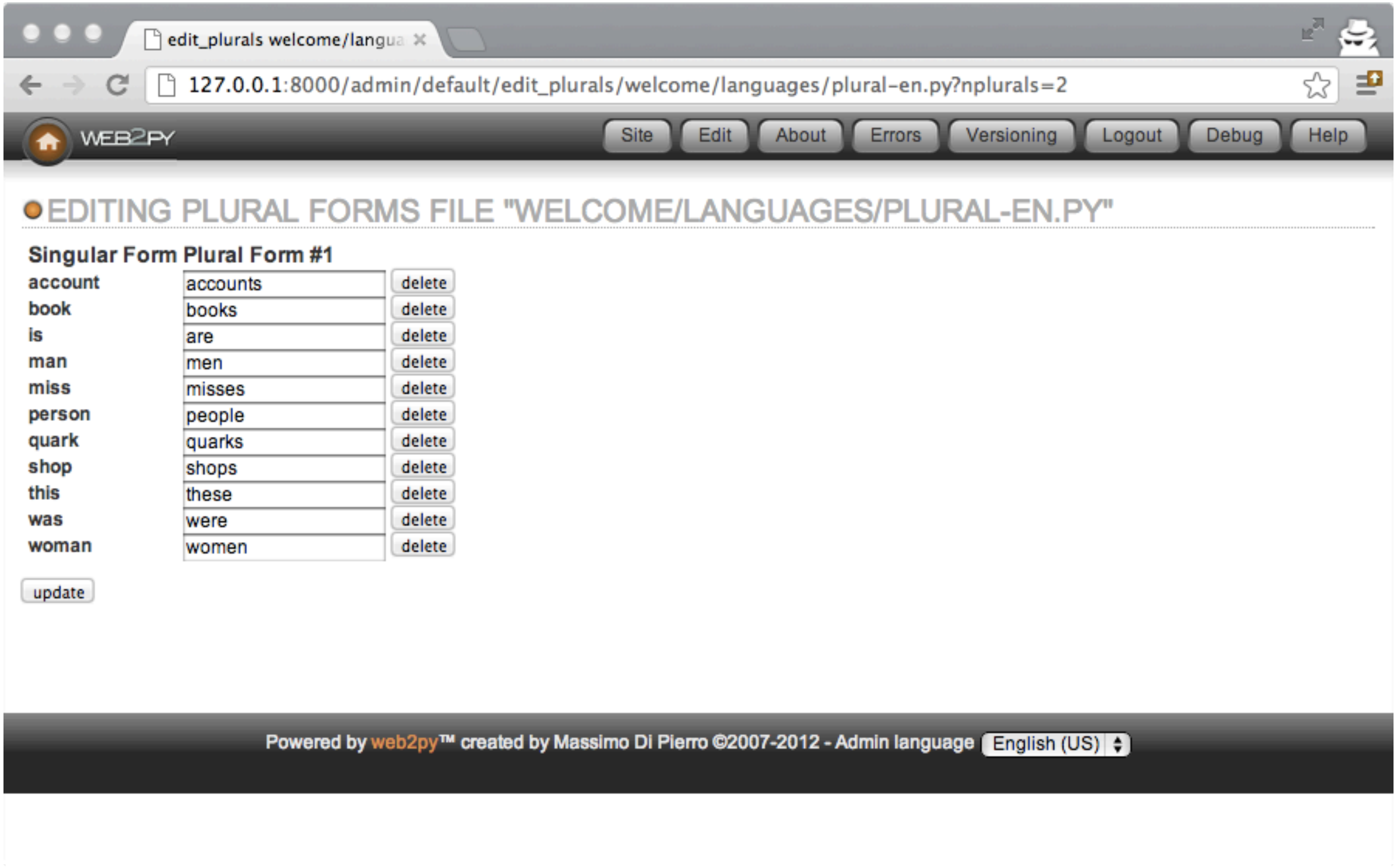
"update" is an optional expression like "field1='newvalue'". You cannot update or delete the results of a JOIN
"update" è un'espressione opzionale come "campo1='nuovo valore'". Non si può fare "update" o "delete" dei risultati di un JOIN delete

%(nrows)s records found
%(nrows)s records found delete

%d seconds ago
%d seconds ago delete

%s %%{row} deleted

Built-in pluralization system

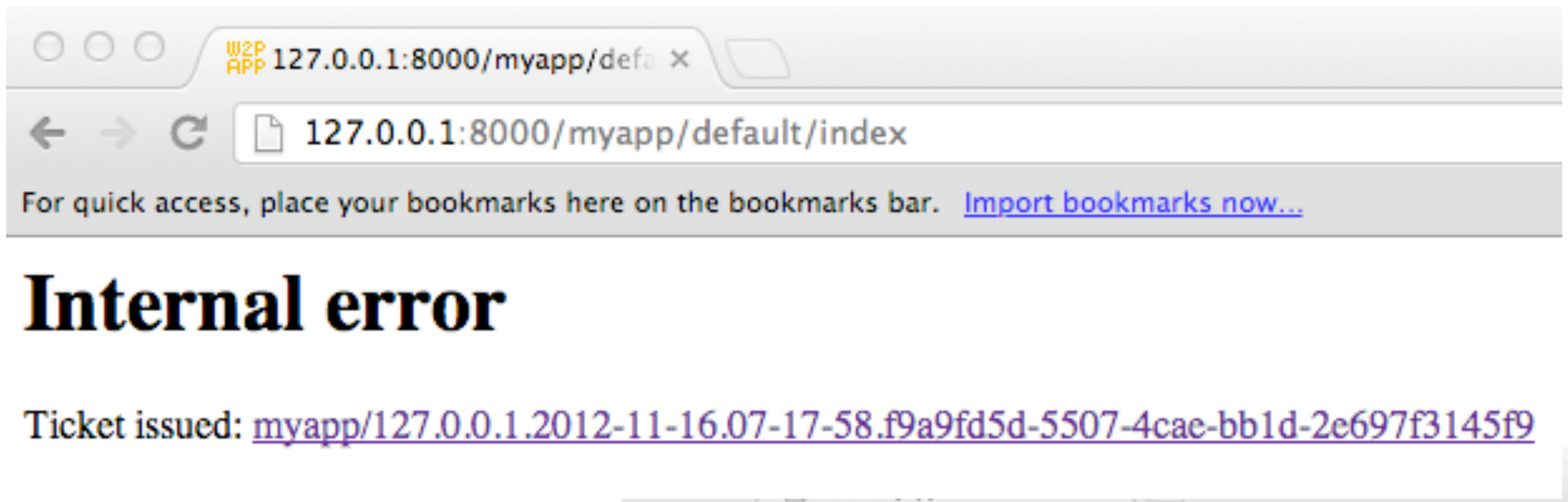

edit_plurals welcome/langua
127.0.0.1:8000/admin/default/edit_plurals/welcome/languages/plural-en.py?nplurals=2
WEB2PY
Site
Edit
About
Errors
Versioning
Logout
Debug
Help

EDITING PLURAL FORMS FILE "WELCOME/LANGUAGES/PLURAL-EN.PY"

Singular Form	Plural Form #1	
account	accounts	delete
book	books	delete
is	are	delete
man	men	delete
miss	misses	delete
person	people	delete
quark	quarks	delete
shop	shops	delete
this	these	delete
was	were	delete
woman	women	delete

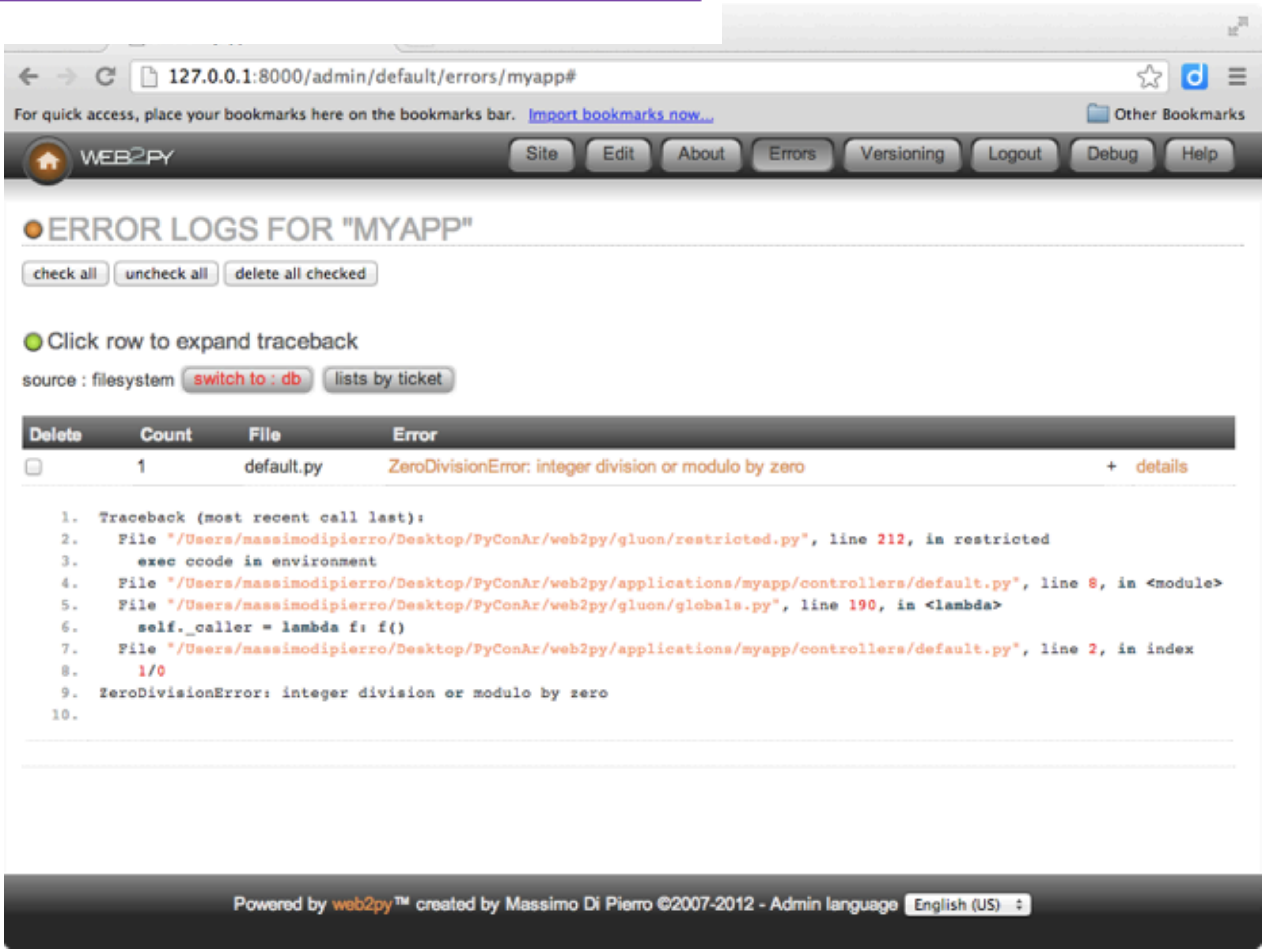
update
Powered by web2py™ created by Massimo Di Pierro ©2007-2012 - Admin language English (US)

Built-in ticketing system


127.0.0.1:8000/myapp/defa
127.0.0.1:8000/myapp/default/index
For quick access, place your bookmarks here on the bookmarks bar. Import bookmarks now...

Internal error

Ticket issued: myapp/127.0.0.1.2012-11-16.07-17-58.f9a9fd5d-5507-4cae-bb1d-2e697f3145f9



ERROR LOGS FOR "MYAPP"

Click row to expand traceback

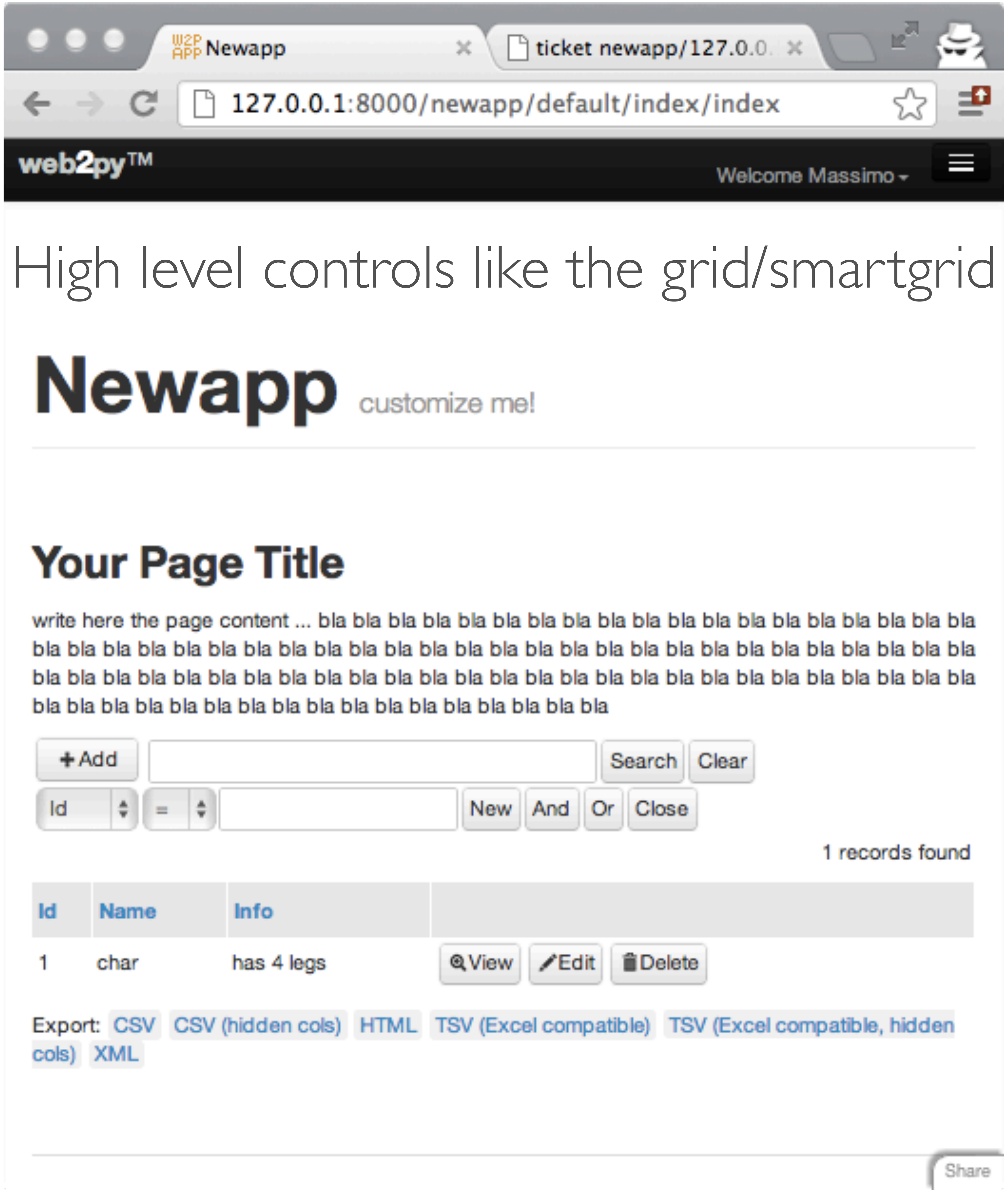
source : filesystem

Delete	Count	File	Error		
☐	1	default.py	ZeroDivisionError: integer division or modulo by zero	+	details

```
Traceback (most recent call last):
  File "/Users/massimodipierro/Desktop/PyConAr/web2py/gluon/restricted.py", line 212, in restricted
    exec ccode in environment
  File "/Users/massimodipierro/Desktop/PyConAr/web2py/applications/myapp/controllers/default.py", line 8, in <module>
  File "/Users/massimodipierro/Desktop/PyConAr/web2py/gluon/globals.py", line 190, in <lambda>
    self._caller = lambda f: f()
  File "/Users/massimodipierro/Desktop/PyConAr/web2py/applications/myapp/controllers/default.py", line 2, in index
    1/0
ZeroDivisionError: integer division or modulo by zero
```

Powered by web2py™ created by Massimo Di Pierro ©2007-2012 - Admin language

High level controls like the grid/smartgrid



Newapp customize me!

Your Page Title

write here the page content ... bla bla bla bla bla bla bla bla bla bla bla bla bla bla bla bla bla bla bla bla bla bla bla bla bla bla bla bla bla bla bla bla bla bla bla bla bla bla bla bla bla bla bla bla bla bla bla bla bla bla bla bla bla bla bla bla bla bla bla bla bla bla bla bla bla bla bla bla bla bla bla bla bla bla bla bla bla bla bla bla bla bla bla bla bla bla bla bla bla

1 records found

Id	Name	Info	
1	char	has 4 legs	View Edit Delete

Export: CSV CSV (hidden cols) HTML TSV (Excel compatible) TSV (Excel compatible, hidden cols) XML

SYNTAX

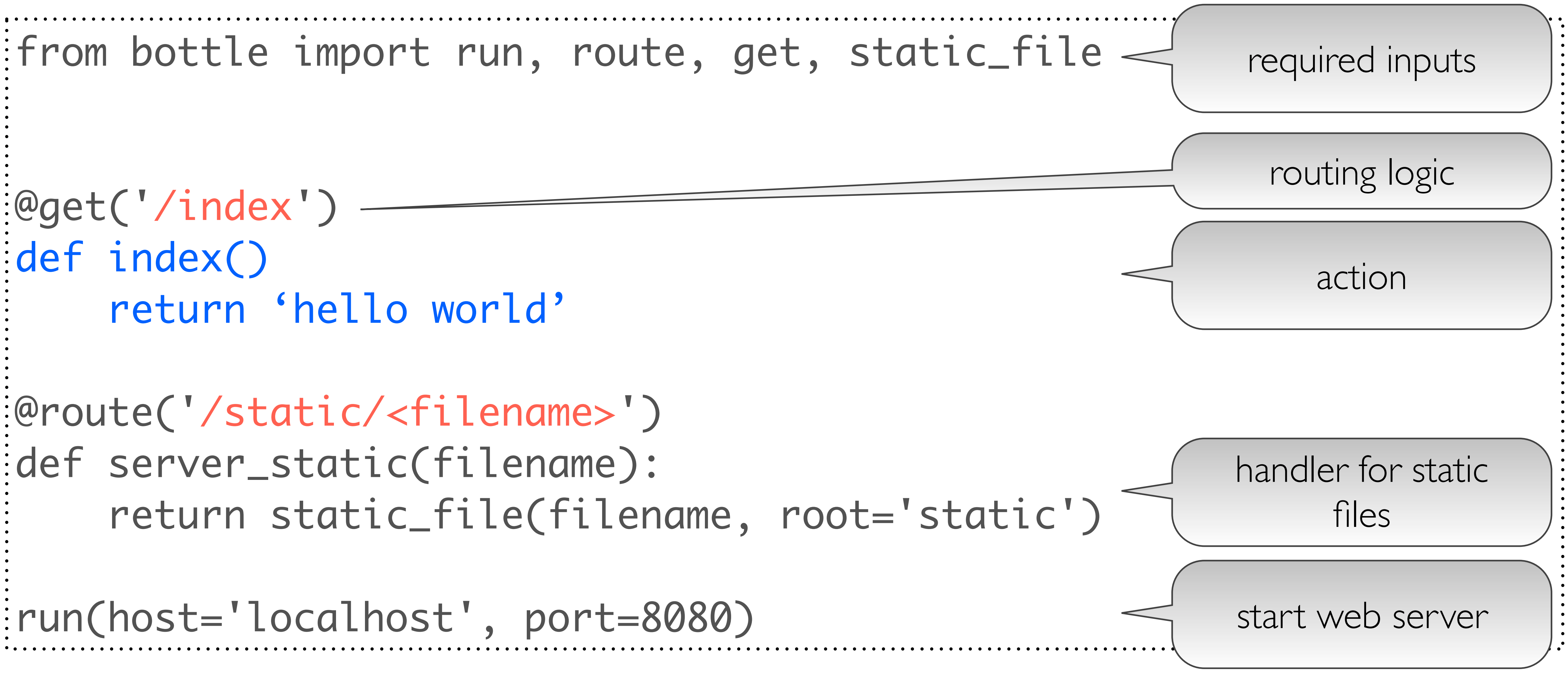
```
public class HelloWorld {
    public static void main(String[] args) {
        System.out.println("Hello, World");
    }
}
```

VS

```
print “hello world”
```

KEEP NEW PROGRAMMERS IN MIND

BOTTLE EXAMPLE



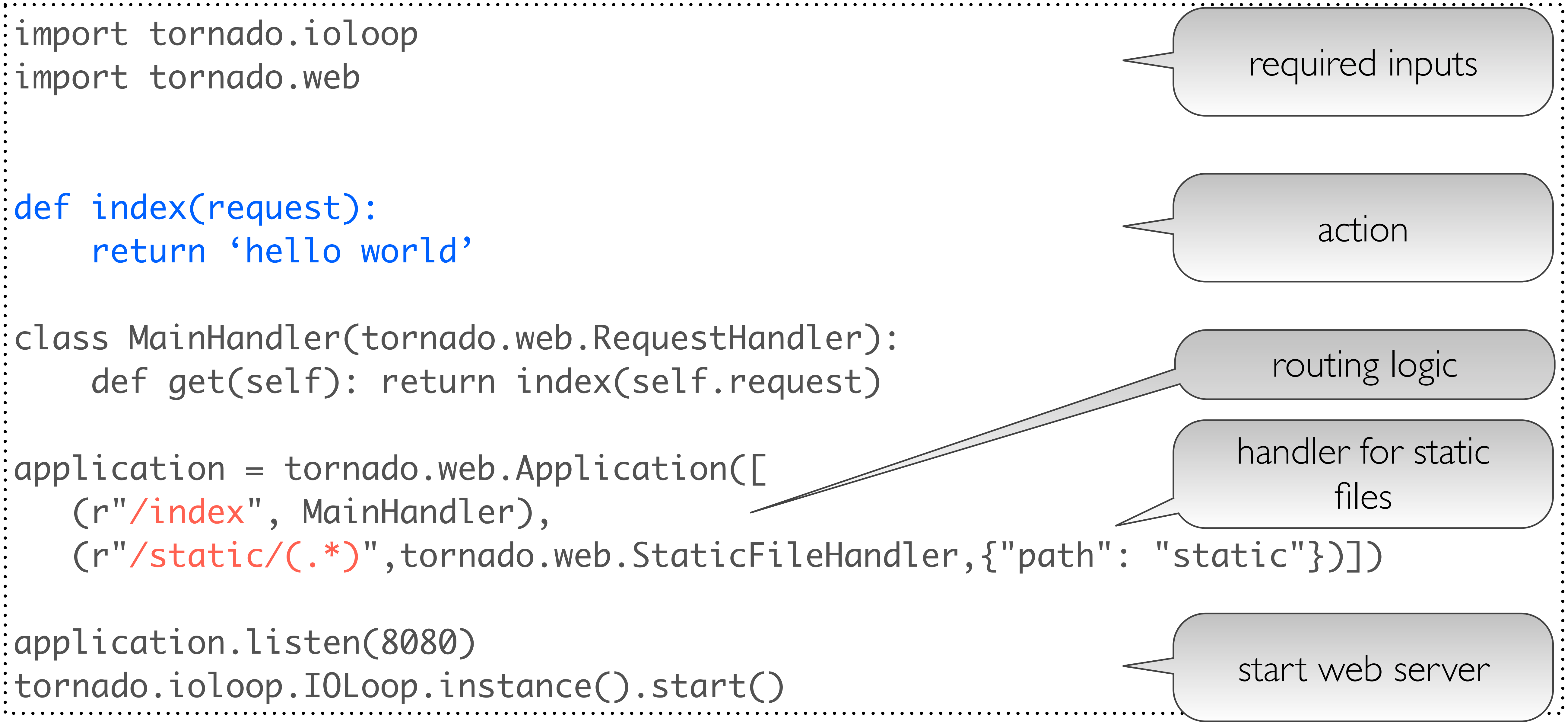
```
from bottle import run, route, get, static_file


@get('/index')
def index()
    return ‘hello world’

@route('/static/<filename>')
def server_static(filename):
    return static_file(filename, root='static')

run(host='localhost', port=8080)
```

FLASK EXAMPLE

```
from flask import Flask, request



app = Flask(__name__)
app.config.from_object(__name__)

@app.route('/index',methods=['GET'])
def index()
    return ‘hello world’

app.run(port=8080)
```

- required inputs
- boilerplate config logic
- routing logic
- action
- start web server

TORNADO EXAMPLE



```
import tornado.ioloop
import tornado.web


def index(request):
    return ‘hello world’

class MainHandler(tornado.web.RequestHandler):
    def get(self): return index(self.request)

application = tornado.web.Application([
   (r"/index", MainHandler),
   (r"/static/(.*)",tornado.web.StaticFileHandler,{"path": "static"})])

application.listen(8080)
tornado.ioloop.IOLoop.instance().start()
```

PYRAMID EXAMPLE

```
from wsgiref.simple_server import make_server
from pyramid.config import Configurator
from pyramid.response import Response
from pyramid.static import static_view

def index(context, request):
    return Response(‘hello world’)

config = Configurator()
config.add_route('index', '/index')
config.add_view(index, route_name='index')
config.add_static_view(name='static', path='static')
app = config.make_wsgi_app()
server = make_server('0.0.0.0', 8080, app)
server.serve_forever()
```

required inputs

action

routing logic

handler for static files

start web server

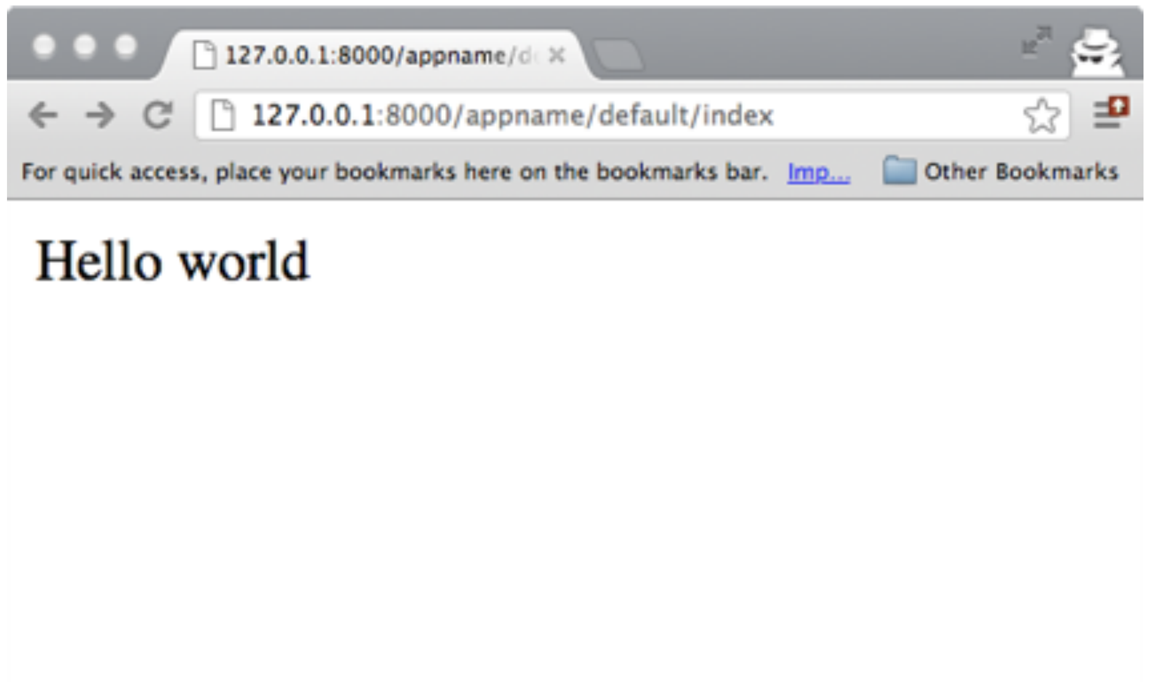
WEB2PY EXAMPLE

http://127.0.0.1:8000/appname/default/index

call

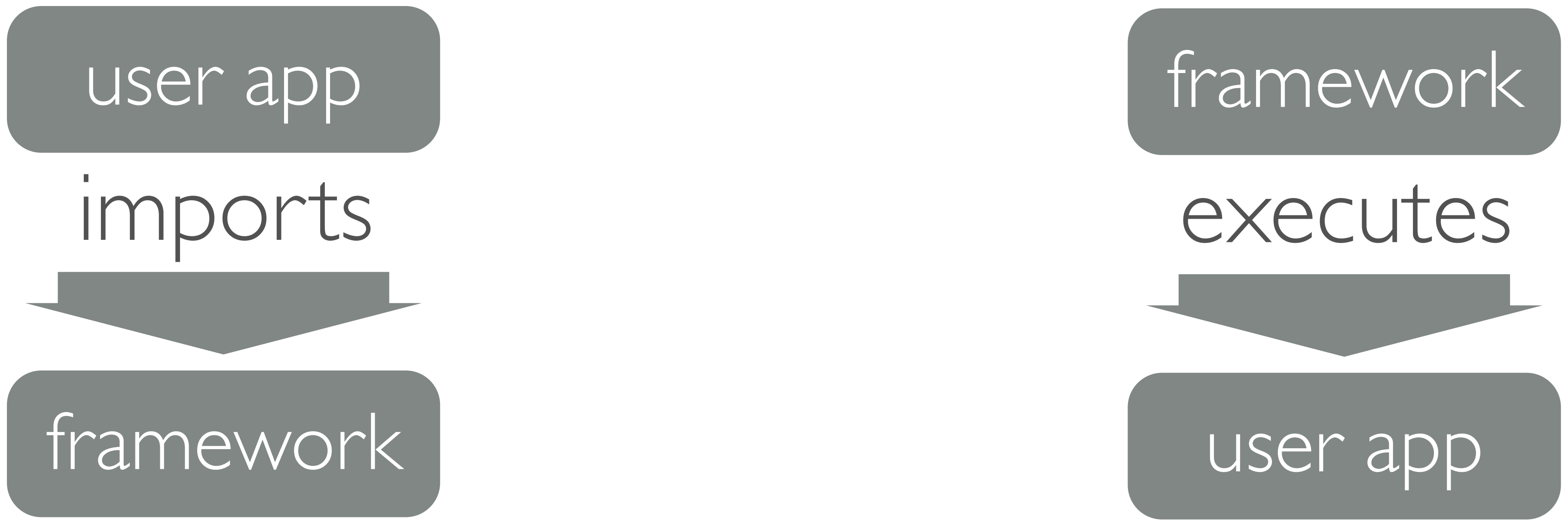
```
def index():
    return ‘hello world’
```

action

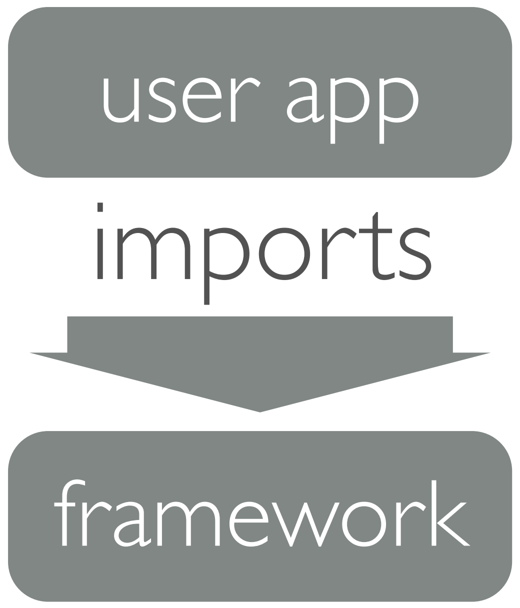

127.0.0.1:8000/appname/d
127.0.0.1:8000/appname/default/index
For quick access, place your bookmarks here on the bookmarks bar. Imp...
Other Bookmarks

Hello world

IMPORT VS EXEC

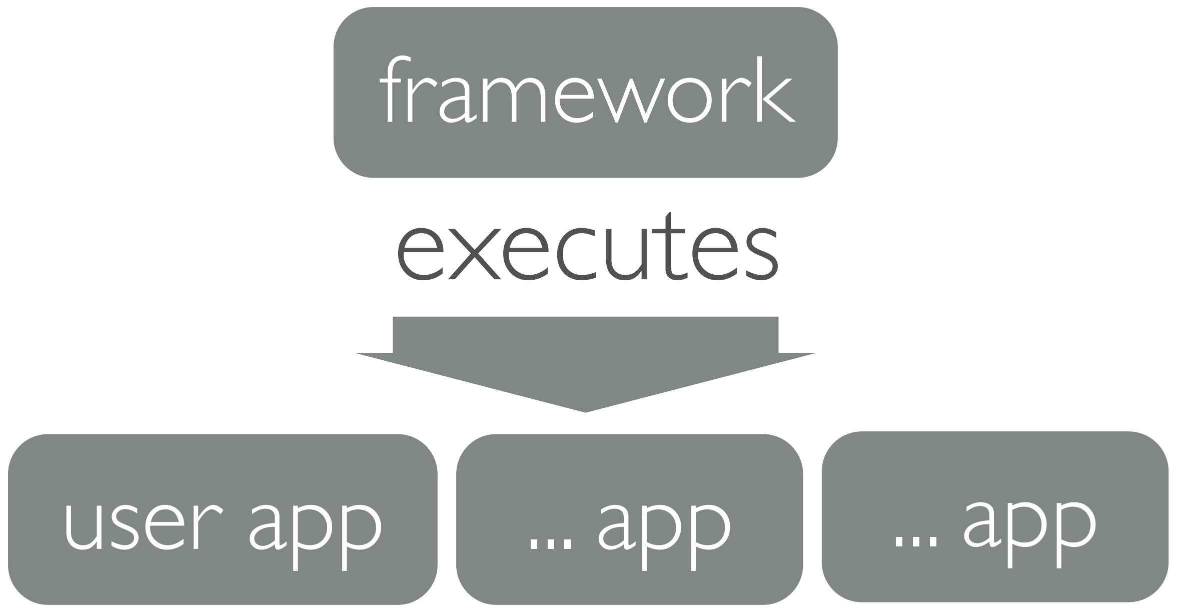


IMPORT VS EXEC



```
from bottle import ...
from flask import ...
from tornado import ...
from pyramid import ...
```

explicit better than implicit

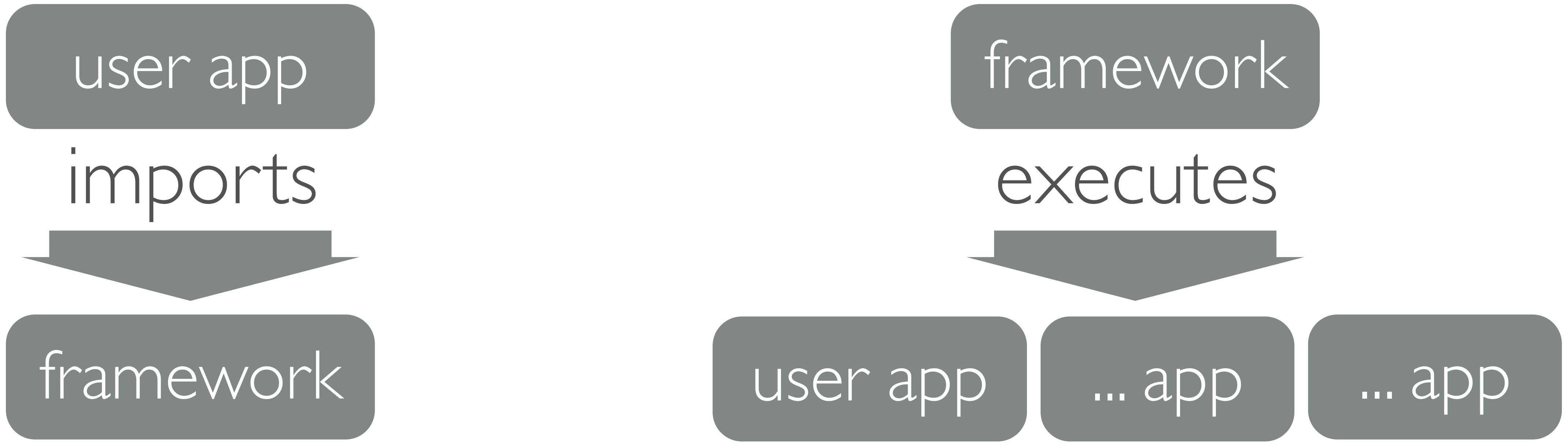


```
env = build_environment(request)
app = find_application(request)
exec app in env    (oversimplification)
```

do not repeat yourself

convention over configuration

IMPORT VS EXEC



faster (for simple apps)
more flexibility
no “magic”

less code (for simple apps)
how swap of code
multi app/multi project
homogeneous environment
“magic”

LAYERS OF CODE

SQL inside Python

```
execute('select * from users where id=1')
```

(DAL or ORM)

```
db(db.users.id==1).select()
```

HTML inside CODE

```
return '<div><h1>%s</h1></div>' % x
```

(helpers)

```
return DIV(H1(x))
```

CODE in HTML

```
<div>{{if x}}check{{endif}}</div>
```

(MVC)

```
<div>{{if x:}}check{{pass}}</div>
```

JS in HTML

```
<div><script>alert('hi!')</script></div>
```

```
<div>{{=LOAD('action',ajax=True)}}</div>
```

WEB2PY DAL

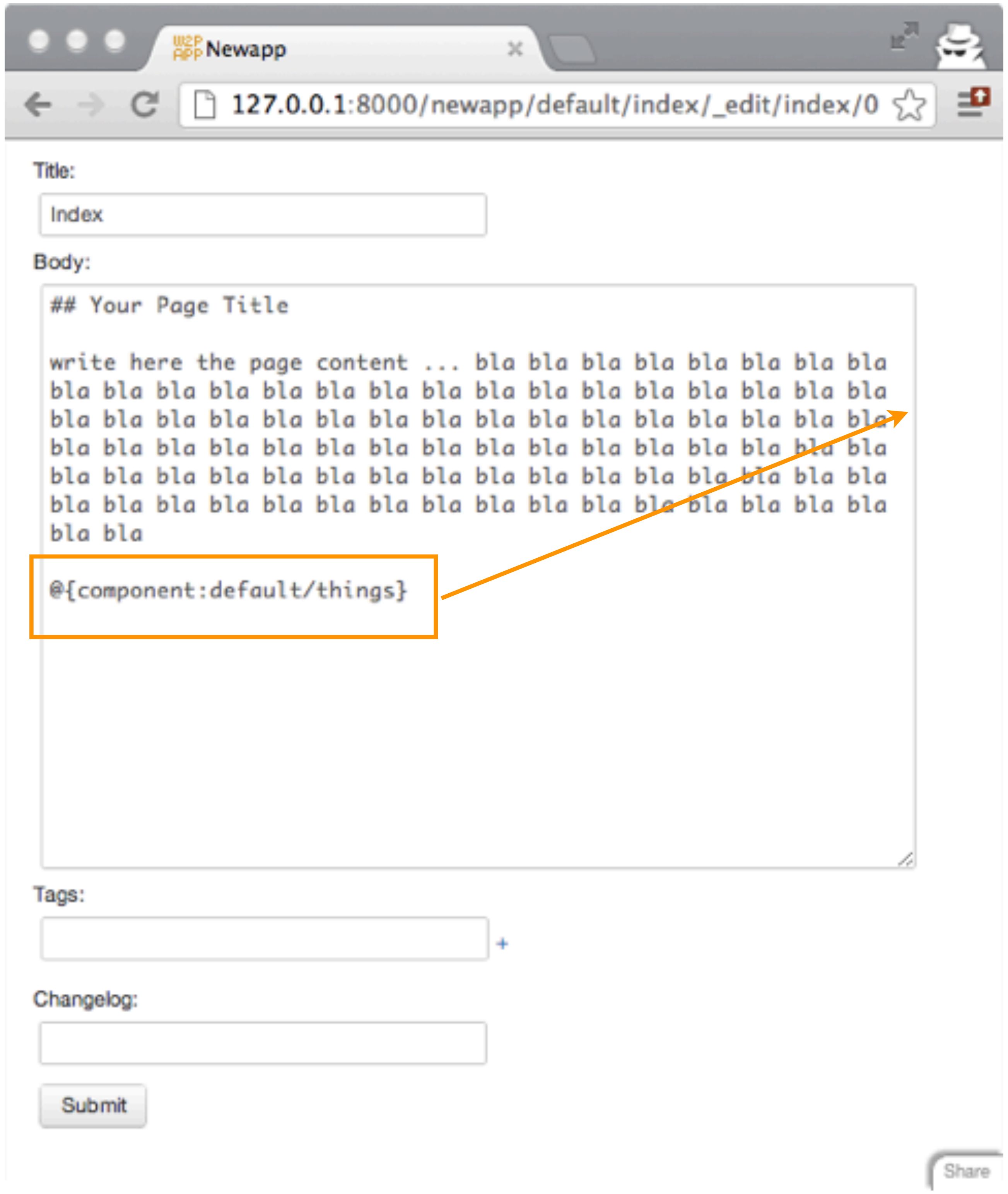
- SQLite, MySQL, PotsgreSQL, MSSQL, Firebid, Oracle, DB2, Ingres, Informix, Ingres, Sybase, GAE, ...
- automatic migrations
- multiple dbs, connection pooling, Round Robin redundancy, distributed transactions
- joins, left joins, aggregates, nested selects, recursive selects

```
db = DAL('postgresql:...', pool_size=10)
db.define_table('person',Field('name'))
db.define_table('thing',Field('name'),Field('owner',db.person))
db.thing.insert(name='PC', owner=db.person.insert(name='John'))

ownership = (db.person.id == db. thing.owner)
thing_counter = db.thing.id.count()
rows = db(ownership).select(db.person.name, thing_counter,
                                           groupby= db. person.id)

for row in rows: print row.person.name, row(thing_counter)
```

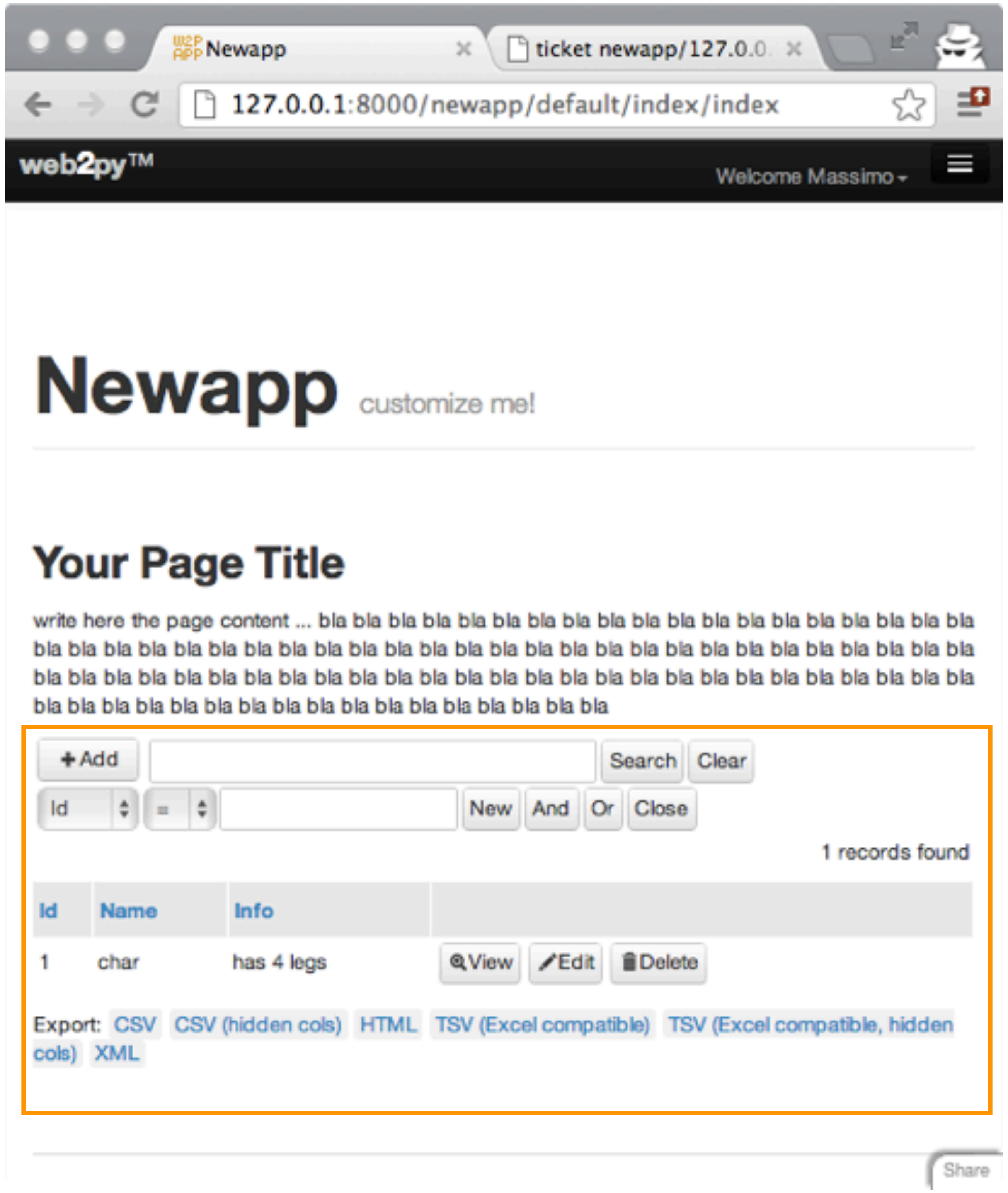
PROGRAMMING AS WIKI

```
# models/db.py
db.define_table('thing',
    Field('name'),
    Field('info','test'))

# controllers/default.py
def index():
    return auth.wiki()

def things():
    return SQLFORM.grid(db.thing)
```

PROGRAMMING AS WIKI



```
## Your Page Title

write here the page content ... bla bla bla bla bla bla bla bla
bla bla bla bla bla bla bla bla bla bla bla bla bla bla bla bla
bla bla bla bla bla bla bla bla bla bla bla bla bla bla bla bla
bla bla bla bla bla bla bla bla bla bla bla bla bla bla bla bla
bla bla bla bla bla bla bla bla bla bla bla bla bla bla bla bla
bla bla bla bla bla bla bla bla bla bla bla bla bla bla bla bla
bla bla

@{component:default/things}
```



Newapp customize me!

Your Page Title

write here the page content ... bla

1 records found

Id	Name	Info	
1	char	has 4 legs	View Edit Delete

Export: CSV CSV (hidden cols) HTML TSV (Excel compatible) TSV (Excel compatible, hidden cols) XML

CONCLUSIONS

- We need to build a society where people understand and control technology, not vice versa.
- We need to build tools that are easy to use to allow more people to use technology for the public good
- web2py is one of such tools
- web2py reduces entry barrier to web programming
- web2py reduces maintenance costs for large projects