

Python with Cloud for Startup

Mail : {{ sigmadream@gmail.com , sangkon.han@nixgon.com }}

Blog : {{ hansangkon.tistory.com, sigmadream.wordpress.com }}

목차

- 스타트업, 선택의 순간
- 쉽고 빠른 개발을 위한 노력
- 결론

A close-up photograph of two human hands, palms up, holding small, translucent capsules. The left hand holds a red capsule, and the right hand holds a green capsule. The background is dark and out of focus, with some green foliage visible on the right side. The lighting is soft, highlighting the texture of the skin and the color of the capsules.

스타트업, 선택의 순간

스타트업!! 꿈, 열정, 희망, 그리고...



실상은 그렇지 않다. 그리고 그럴 수 없다.



{ {왜냐하면} }

“누구나 처음”

{{스타트 업의 특징}}

경험 부족, 기술 부족,
코치 부족, 투자 부족

{{그냥 다 부족}}

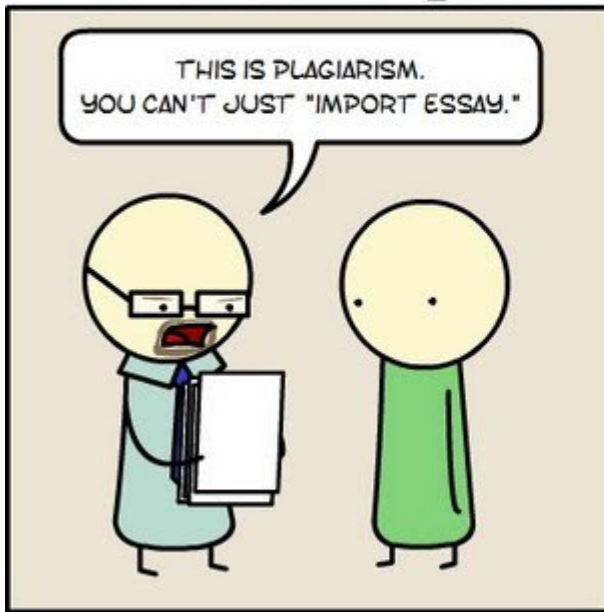
그래서, 살아남기 위한 전략
‘빠른 실패’ 혹은 ‘빠른 수정’

개발팀의 첫번째 전쟁 : 속도

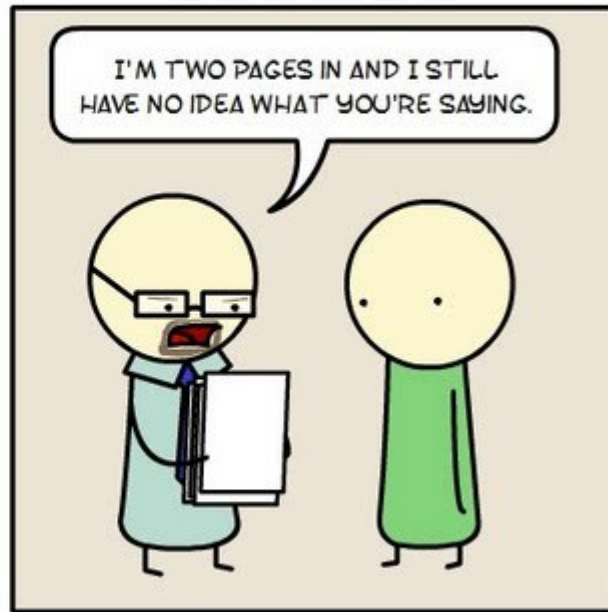


Java v.s. Python

PYTHON



JAVA



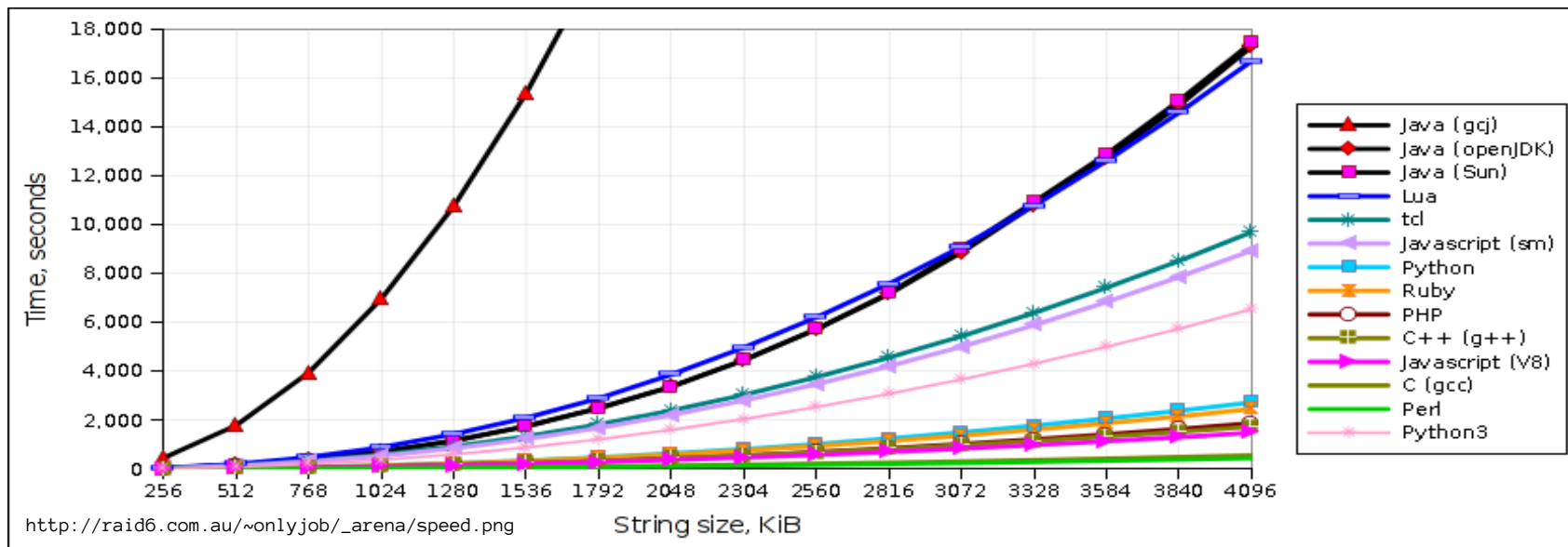
{ { 모든 사람이} } 말하길

“파이썬은 인터프리터 언어(!)라서
속도가 (많이) 느리다.”



아...
이건 정말
듣도 보도 못한
발상이군.

{{신념과 믿음, 혹은 사랑과 우정 그 ‘사이’}}



문명, FM, 그리고 LOL 그 중 제일은 WOW!



테스트 시작 : “잘 만들어봐, 세상의 모든 게시판을
거기에 두고 왔으니!”



JSP v.s. WebApp2"



압도적 차이...

```
- import java.util.Date;

import javax.jdo.annotations.IdGeneratorStrategy;
import javax.jdo.annotations.PersistenceCapable;
import javax.jdo.annotations.IdentityType;
import javax.jdo.annotations.Persistent;
import javax.jdo.annotations.PrimaryKey;

@PersistenceCapable(identityType = IdentityType.NONE)
public class Article {

    @PrimaryKey
    @Persistent(valueStrategy = IdGeneratorStrategy.IDENTITY)
    private Long id;

    @Persistent
    private String title;

    @Persistent
    private String name;

    @Persistent
    private String content;

    @Persistent
    private Date createDate;

    public String getTitle() {
        return title;
    }
}
```

```
import os
import cgi
from datetime import date
from google.appengine.ext import webapp
from google.appengine.ext.webapp import template
from google.appengine.ext.webapp.util import run_wsgi_app
from google.appengine.ext import db
```

```
class Message(db.Model):
    name = db.StringProperty()
    email = db.StringProperty()
    msg = db.StringProperty(multiline=True)
    postedon = db.DateTimeProperty(auto_now_add=True)
```

```
class MyPage(webapp.RequestHandler):
    def get(self):
        messages = Hello!
        template_values = {
            'messages': messages
        }
        path = os.path.join(os.path.dirname(__file__), 'index.html')
        self.response.out.write(template.render(path, template_values))
```

```
def post(self):
    self.redirect('/')
■
```

자신과의 싸움에서 승리한 Python



긴~~~ 코드는 긴~~~ 버그를 동반한다.

9/9

0800 Antan started
 1000 " stopped - antan ✓
 1300 (032) MP - MC ~~1.982147000~~
 (033) PRO 2 2.130476415
 connect 2.130676415

{ 1.2700 9.037847025
 9.037846995 connect
 4.615925059(-2)

Relays 6-2 in 033 failed special speed test
 in relay " 10.000 test -

1100 Started Cosine Tape (Sine check)
 1525 Started Multi Adder Test.

1545



Relay #70 Panel F
 (moth) in relay.

1630 antan started.
 1700 closed down.

First actual case of bug being found.

Relay
 2145
 Relay 3370

{{소소한 문제(들)}}}

- 약형Weakly-typed Language 언어
- 버전 충돌
- 디버깅
- 개발자 구인

{{해결책 #1}}

“타입검사보다는 테스트에 더
공을 들이자” – Bruce
Eckel

{ {해결책 #2} }

Virtualenv

{{해결책 #3}}

- Unittest

✓ <https://sites.google.com/site/tddproblems/>

그리고
회사에
그분이
오셨다

본격 'Vi(m)'과 'Emacs' 등장

```
" :ClearBookmarks [<names
" :ClearAllBookmarks

.. (up a dir)
/+ $RECYCLE.BIN/
/+ bla/
/+ boost/
/+ Classic/
/~ CodeBlocks/
/~ CMake28/
/~ MinGW/
/~ bin/
/~ doc/
/~ include/
/~ lib/
/~ libexec/
/~ mingw32/
/~ msys/
/~ share/
/~ var/
/~ cache/
/~ mingw-get/
/~ data/
/~ packages/
/~ gdb-7.5-1-m
/~ make-3.82.9
/~ lib/
/~ log/
/~ manifest-0-030-38
/~ manifest-0-030-b9
/~ manifest-0-030-cb
/~ manifest-0-032-0b
/~ manifest-0-032-2f
/~ manifest-0-032-e1
/~ manifest-0-033-91
/~ manifest-0-033-c3
/~ manifest-0-034-bb
/~ manifest-0-034-c1

pathogen#glob_directories 383
pathogen#cycle_filetype 384
pathogen#is_disabled 385
pathogen#runtime_prepend_subdir 386
pathogen#runtime_append_all_bun 387
pathogen#helptags 388
pathogen#runtime_findfile 389
pathogen#fnameescape 390
find 391
Findcomplete 392

imgconv.py (D:\foe\edscii_alpha4) 393
class 394
  LabPalette 395
  StringCharmap 396
  StringChar 397
  ImageConverter 398
member 399
  __init__ [LabPalette] 400
  closest_color [LabPalette] 401
  rgb_colors [LabPalette] 402
  dominant_colors [LabPalette] 403
  __init__ [StringCharmap] 404
  __init__ [StringChar] 405
  __repr__ [StringChar] 406
  get_opaque_pixels [StringChar] 407
  __cmp__ [StringChar] 408
  __init__ [ImageConverter] 409
  tick [ImageConverter] 410
  prepare_image [ImageConverter] 411
  color_reduce [ImageConverter] 412
  convert_step [ImageConverter] 413
  get_block [ImageConverter] 414
  best_fit_char [ImageConverter] 415
  block_diff [ImageConverter] 416
  finished [ImageConverter] 417
  abort [ImageConverter] 418

len(self.pal.colors), ('', ' (with dither)')[self.dither])
self.app.msgline.set_string(txt)
# stop here, let app tick and move to next step

def tick(self):
    # check step we're at
    if self.state == IC_STARTING:
        self.state = IC_IMAGE_PREP
    elif self.state == IC_IMAGE_PREP:
        self.prepare_image()
    elif self.state == IC_COLOR_REDUCE:
        self.color_reduce(self.source)
        self.app.msgline.set_string('starting conversion...', 3000)
    elif self.state == IC_BLOCK_CONVERT:
        self.convert_step()
    elif self.state == IC_FINISHED:
        self.finished()

def prepare_image(self):
    NORMAL imgconv.py +
    let j += 1
    endwhile
    else
    let path .= "," . a:000[i]
    endif
    let i += 1
    endwhile
    return substitute(path, '^,', '', '')
endfunction " }}}1

" Convert a list to a path with escaped spaces for 'path',
' tag', etc.
function! pathogen#legacyjoin(...) abort " {{{1
    return call('pathogen#join',[1] + a:000)
endfunction " }}}1

" Remove duplicates from a list.
function! pathogen#uniq(list) abort " {{{1
    let i = 0
    while i < len(a:list)
        while i < len(a:list)
            if a:list[i] == a:list[i+1]
                a:list[i] = ''
            else
                i++
            endif
        endwhile
        i++
    endwhile
    return a:list
endfunction " }}}1
```



쉽고 빠른 개발(같은건 없더라구요!)

쉽고 빠른 개발을 위한 팁(!)

- 좋은 IDE와 개발환경
- 강력한 개발서버
- 좋은 팀원(!)

좋은 IDE



계속 누군가 scv라 부르는 형상관리 도구! (우리팀은 Yobi!)

Search current project



8 웹 브라우저 패키지 설치 및 설정

한상곤 7 days ago



7 SQL Builder [phrase #1]

한상곤 7 days ago

yobi

협업 개발 플랫폼

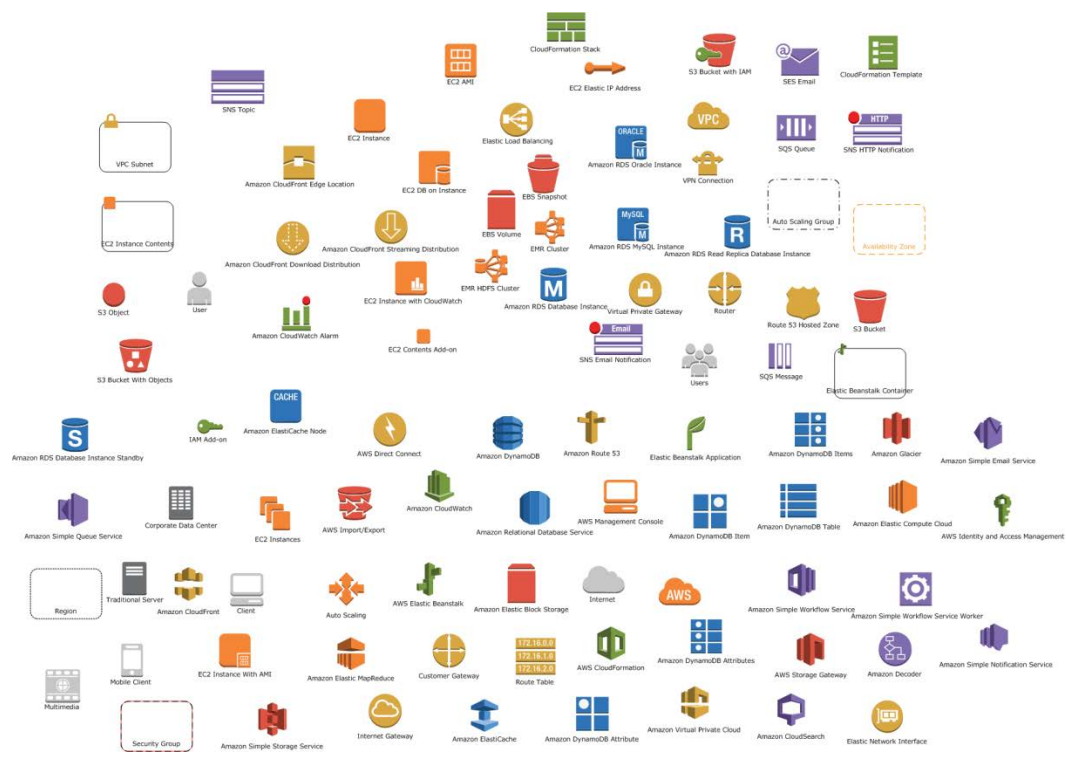
~~강력한 개발서버(!) (가... 이거지 이거!)~~



원지 잘 모르겠지만, 가자 클라우드로!



그렇지만 이걸... 이걸 너무 강력하다...



우리가 있을 곳이 아닌 것 같아요...



그래서 여기로...(저렴, 파이썬 친화적 Go가 기대됨!)



Google
App Engine



우리가 레진 코믹스 직원같이 느껴지던 나날들...

왜
레진코믹스는
구글 앱엔진을 선택했나



강소리
레진엔터테인먼트, GDG서울운영자

클라우드 사용시 주의사항

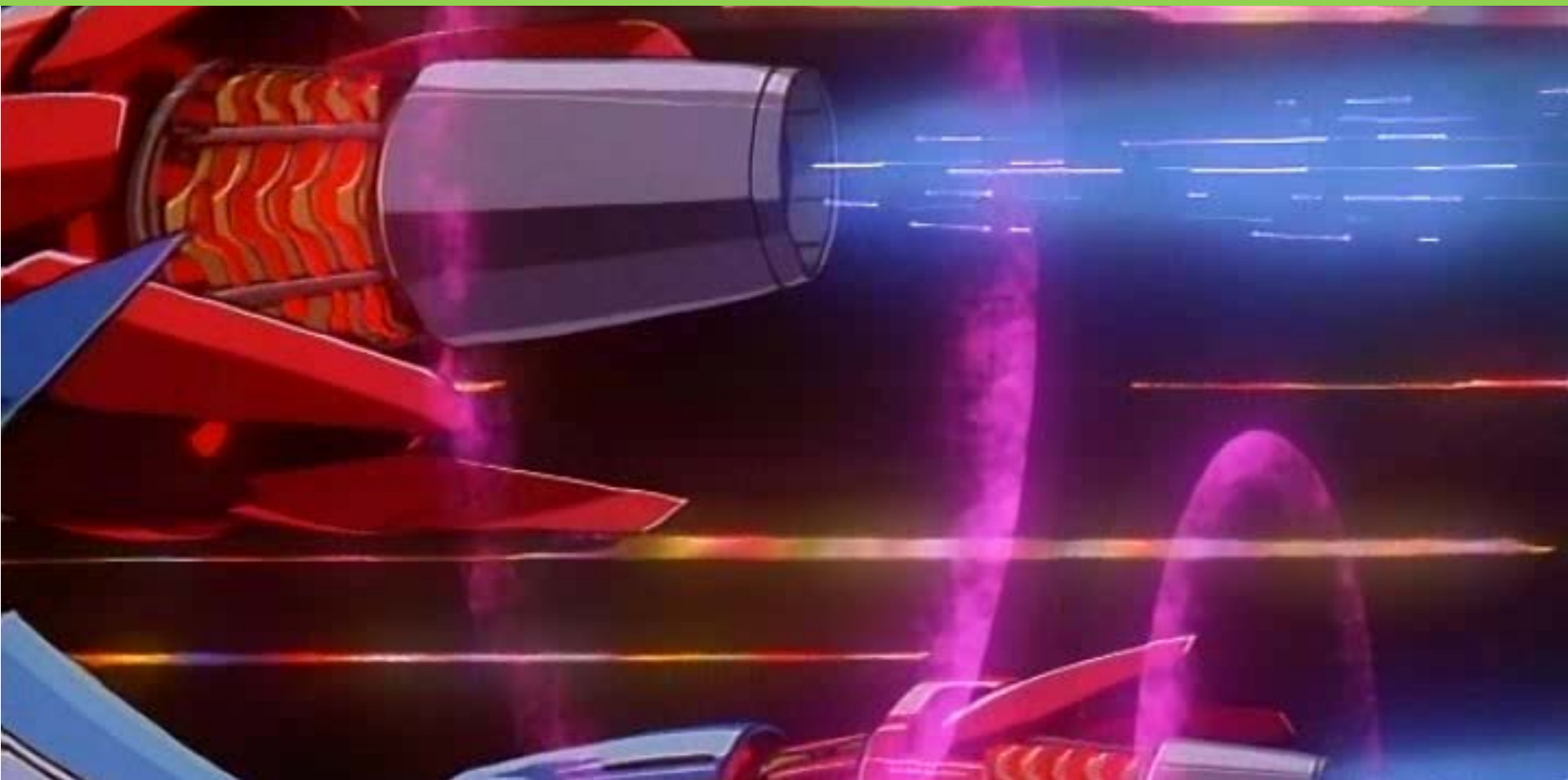
- IaaS >>> PaaS >>> SaaS
- 분산처리, 빅데이터, 맵리듀스 여튼 뭔가 있어보이는 모든 기술은 Python에서 사용 가능
 - 그런데... 생각보다 쉽지 않음. 왜일까?

(여튼) 모든 것이 평화
로울 것이라 생각했던 그
순간...

스프링의 역습



훗... 우리도 있다. 비장의 카드가!



스프링만큼 좋다!

THE D
IS
SILENT.



와이어프레임(혹은 목업)의 최강자!



Flask

web development,
one drop at a time

{{ 믿고 보는 ‘뉴스룸’,
그 뉴스룸을 만든 ‘쟁고’ }}

B.U.T.



개발 2달이 넘어갈 무렵



{ {(또 다시) 사소한 문제들} }

- 클라우드 제약사항
- 프레임워크에 대한 몰이해
- 된다고 믿었던 것들이 안될
때의 멘탈 크리티컬

{ {왜?} }

- 회사에 자바 개발자 비중 80%
- 파이썬 문법은 능숙, 파이썬의 철학과 원리는 반숙

레벨업을 위한 우리의 방법!



Level 90 Character Boost

WORLD OF WARCRAFT® GAME SERVICE

KRW 63,000

Available for purchase in-game [?](#)

- Can be used on a single World of Warcraft character below level 90 (including a newly created character) on a Battle.net account.



{{경험치 획득 방법}}

- Slideshare
- 파이썬 포팅
- 오픈소스 코드 리딩



결론

結論은 버킹엄점입니다

{{파이썬, 강력 추천!}}

- 잘 만들어진 생태계
- 완성도 높은 언어

{{그리고,
이토록 완벽한 커뮤니티!}}

Thanks for Everyone
and
PyCon!

- 사용된 폰트 ‘Nixgon’ (꽤알같은 회사자랑!)
 - <https://github.com/nixgon/NixgonFonts>
- 점프 투 파이썬
 - <https://wikidocs.net/book/1>
- 왜 레진코믹스는 구글앱엔진을 선택했나?
 - http://www.slideshare.net/curioe_/lezhincomics-google-appengine-30453946
- Do not use Django as like as SMARTSTUDY
 - <http://www.slideshare.net/lqez/do-not-use-django-as-like-as-smartstudy>
- RailsConf 2010: Robert Martin
 - <http://www.youtube.com/watch?v=ms1MLp5bQD0&feature=youtu.be&t=35m24s>