

La importancia de los
proyectos de
código abierto en la
formación profesional

Linux Week 2010

Lima

¿Por que están aquí?

Respuestas que no valen

Para aprender

Para ver que dicen...

¿Cuál es el perfil de un informático?

En la PUCP

Analiza los procesos
de negocio
y las necesidades
de información de la
organización.

En la PUCP

Ejecuta y administra las actividades del ciclo de vida de proyectos informáticos, utilizando tecnologías, estándares y herramientas adecuadas.

En la PUCP

Planifica, implementa, configura
y mantiene la
infraestructura informática
de la organización.

En la PUCP

Se desempeñará como analista de sistemas de información, administrador de proyectos informáticos, administrador o consultor de tecnologías de información, desarrollador de software, auditor de sistemas de información, investigador, entre otros.

¿Cuál es tu objetivo?

¿Qué esperas obtener
de tu carrera?

¿En que trabajarás?

¿Tienes los conocimientos
necesarios?

Las asignaturas típicas de
la carrera.

Primer Semestre

Algoritmos básicos

segundo semestre

Elementos de un Lenguaje estructurado

En el resto de la
carrera

Pilas

Recursión

Listas enlazadas

Colas

grafos

ARBOLES

conjuntos

POO

Ingeniería de software

Métodos ágiles

¿Tienes los conocimientos
necesarios?

¿Sabes aplicarlos?

¿Qué se debe preguntar
En una entrevista de
trabajo?

Escribe una función que
invierta una cadena
abcdefg → gfedcba

Describe un árbol binario

Escribe una función que inserte un número en un árbol binario de números.

¿cómo está tu nivel
de inglés?

The next step should be to edit your test code and add an import of any Django-specific code you're using – your views, your models, your URLconf, your RSS configuration, and so forth. Put these imports in your test handler function and access your test URL in a browser.

¿cómo vas a obtener
los conocimientos
necesarios?

En general la respuesta
es una sola

Iniciativa propia

Interés por aprender

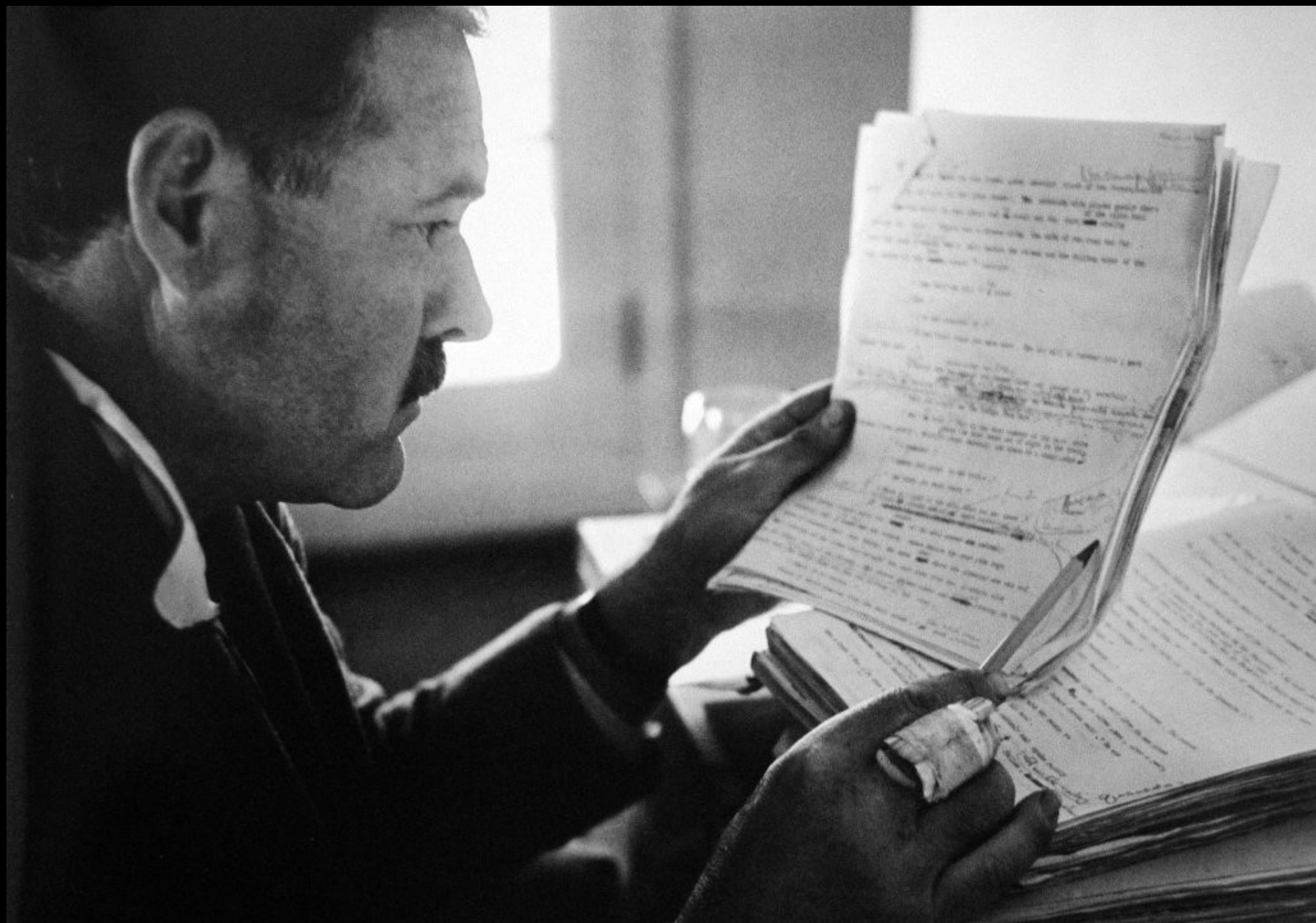
Tener un buen
Nivel de inglés

práctica

práctica

práctica

Mucha práctica



Fernando San Martín Woerner - fsmw@gnome.org

La foto es de 1940

obtuvo el nobel 1954

Las grandes obras
no salen en el
primer intento

¿que tiene que
ver todo esto
con el software libre?

libertad

Libertar para utilizarlo

No tienes que pedir
Permiso a nadie

Puedes usarlo para fines
que el autor no esperaba

Puedes usarlo en tu casa
escuela o trabajo

Solo necesitas iniciativa
propia

Libertad para copiarlo

Dale una copia a tu
Mejor amigo

Dale una copia al chico/a
que te gusta!

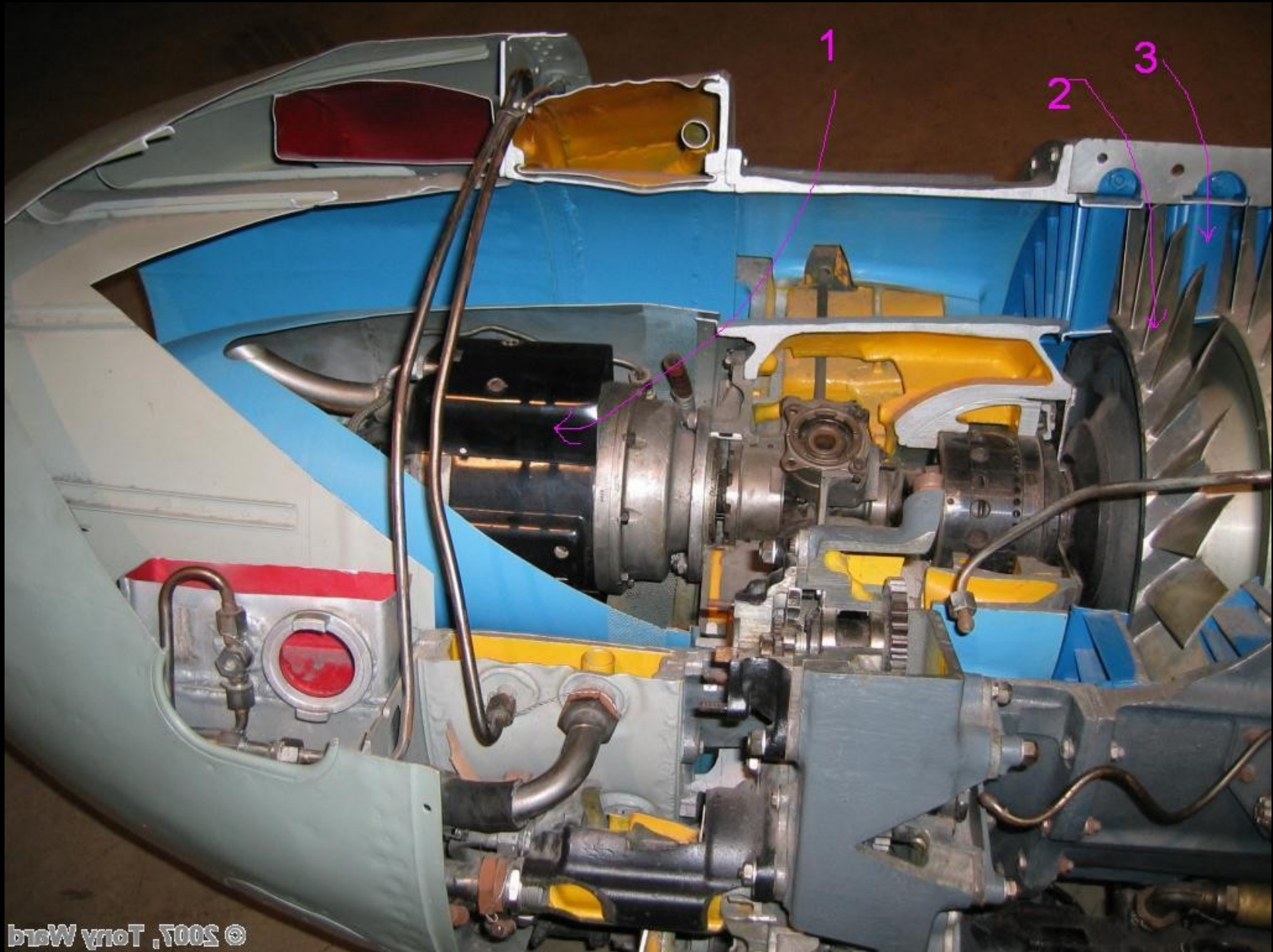
Instalalo en todas tus
computadoras

¡Vende los CD/DVD
con las copias!

Solo necesitas iniciativa
propia

Libertad para modificarlo

¡examinalo!



© 2007 Tony Ward

Aprende como está hecho

Haz cambios

Aprende de lo que hiciste

Sé un mecánico experto

Solo necesitas iniciativa
propia

Libertad para distribuirlo

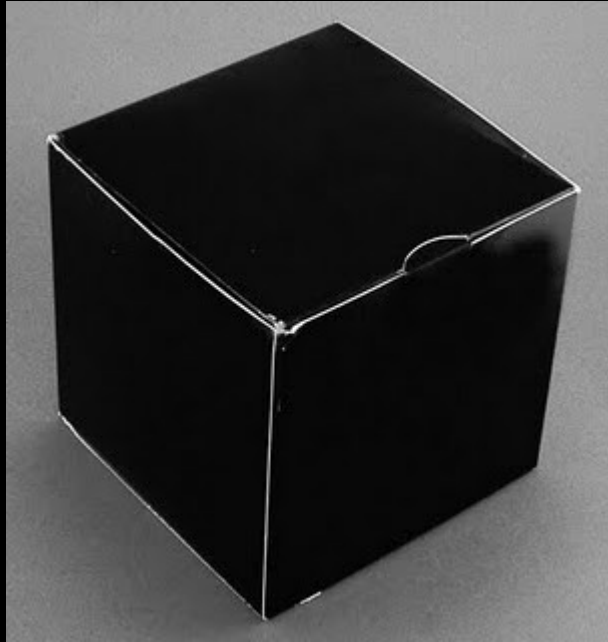
¿Hiciste cambios interesantes?

Vendelos / Distribuyelos / Copialos

Hazlos parte del bien común

Solo necesitas iniciativa
propia

El software propietario es una
Caja negra



El software libre es una
Venta abierta



Aprende de programas reales

Aprende nuevas técnicas

Aprende de expertos de todo
el mundo

Aplica la teoría

Solo necesitas iniciativa
propia

¿Como hago todo eso?

1

Descarga el código fuente

2

Desempaquetado (tar/zip)

3

Construye y ejecuta la aplicación

4

Abre el código fuente en un editor

5

Encuentra las partes del código
que deseas modificar

6

Aplica los cambios hasta que
funcione correctamente

7

Ejecuta el comando `diff -u`
y envías la salida por correo
al desarrollador o lista
correspondiente

Repita los pasos anteriores
todas las veces que sea
necesario

Si no hay programas que
desees modificar

Tal vez no quieras ser un
desarrollador después de todo

Tal vez no has usado el
programa lo suficiente

Tal vez este año solo seas
un usuario

Es imposible saltarse los
pasos del 1 al 4

¿Si no tienes el código fuente
podrías llegar a ser un
desarrollador?

Al menos nadie lo ha logrado
hasta el momento.

El paso 3 es el más frustrante
al inicio

Puede tomar un tiempo hasta
que tengas la aplicación
corriendo

Mientras más experiencia tienes
el paso 5 es más fácil

Los programadores
experimentados escriben
software más fácil de leer

El paso 6 es lo que se conoce
como "hacking"

Pero no es lo que toma más
tiempo

Muchos no llegan a este punto

Si lo has hecho te puedes
llamar "hacker"

Pero es probable que...

No escribas buen código
hasta que hayas programado
un buen tiempo

Nunca es tarde para comenzar

Tienes la libertad de hacerlo
ahora

solo necessitas iniciativa propria

Muchas gracias

Basado en la charla de:
Fedérico Mena y Germán Poo