

MANUAL TÉCNICO

PROTOTIPO DE APLICATIVO WEB PARA LA GESTIÓN DE GRANJA PORCINA DE LA
COOPERATIVA MULTIACTIVA PARA EL BUEN VIVIR Y LA PAZ DEL CAQUETÁ
UBICADA EN LA VEREDA AGUABONITA EN EL MUNICIPIO LA MONTAÑITA

ANDRÉS FELIPE COLLAZOS ROZO

1088332068

UNIVERSIDAD TECNOLÓGICA DE PEREIRA
FACULTAD DE INGENIERÍAS
INGENIERÍA DE SISTEMAS Y COMPUTACIÓN
PEREIRA

2019

MANUAL TÉCNICO
PROTOTIPO DE APLICATIVO WEB PARA LA GESTIÓN DE GRANJA PORCINA DE LA
COOPERATIVA MULTIACTIVA PARA EL BUEN VIVIR Y LA PAZ DEL CAQUETÁ
UBICADA EN LA VEREDA AGUABONITA EN EL MUNICIPIO LA MONTAÑITA

ANDRÉS FELIPE COLLAZOS ROZO

1088332068

MANUAL TÉCNICO ANEXO AL PROYECTO DE GRADO PRESENTADO COMO
REQUISITO PARA LA OBTENCIÓN DEL TÍTULO DE INGENIERO DE SISTEMAS Y
COMPUTACIÓN

CARLOS ALBERTO OCAMPO SEPULVEDA

DIRECTOR DE PROYECTO

UNIVERSIDAD TECNOLÓGICA DE PEREIRA

FACULTAD DE INGENIERÍAS

INGENIERÍA DE SISTEMAS Y COMPUTACIÓN

PEREIRA

2019

Indice

Introducción.....	4
Requisitos del sistema.....	5
Instrucciones de instalación en el entorno local.....	6
Instrucciones de instalación en servicio externo.....	13
Administración de usuarios.....	18
Características de los usuarios.....	21
Definición del desarrollo.....	22
Definición de la implementación: herramientas y librerías.....	23
Vista de implementación o vista física.....	25
Glosario.....	26
Referencias.....	28

Introducción

El presente documento pretende dar a conocer las características técnicas del prototipo de aplicativo web OinkSoft, los requerimientos tecnológicos para el despliegue o puesta en marcha de la aplicación tanto en un entorno local o de desarrollo, como en un entorno de producción. Se especifican las herramientas utilizadas para su desarrollo y las características de las tecnologías implementadas.

Requisitos del sistema

Para el entorno de desarrollo local del prototipo, los requisitos son:

- Sistema operativo Arch Linux (Cualquier sistema operativo Linux funciona)
- 100 MB en disco duro para el aplicativo
- 1 GB de memoria RAM
- Gestor de bases de datos: SQLite
- Editores de texto (opcional): Atom (400 MB)
- Web2py 2.18.5
- Python 3.7.3

Para el entorno de producción:


- PythonAnywhere: como servidor que soporta SQLite y cuenta con un instalador automático del framework, lo cual facilita trabajo. Además, PythonAnywhere nos sirve como hosting de pequeña demanda, ideal para este prototipo.

Instrucciones de instalación en el entorno local


El framework web2py utilizado para el desarrollo del prototipo de aplicativo web, se distribuye como código fuente, y en formato binario para sistemas Microsoft. En este manual, y en este proyecto de grado, nos centraremos en el uso de herramientas con licencia de Software Libre, en ese sentido, haremos uso del sistema operativo GNU/Linux.

Para obtener el framework nos dirigiremos a la página oficial del Web2py


[Home](#) [About](#) [Download](#) [Docs & Resources](#) [Support](#) [Contributors](#)



Free open source full-stack framework for rapid development of fast, scalable, **secure** and portable database-driven web-based applications.
Written and programmable in **Python** (version 3 and 2.7).



Current version:
2.18.5-stable+timestamp.2019.04.08.04.22.03 (LGPLv3 License)



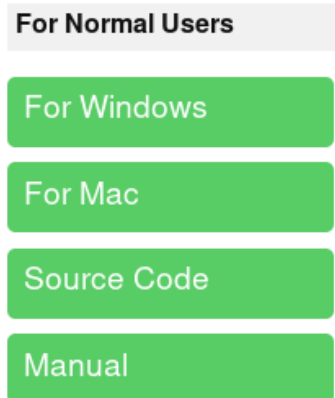
- Download Now
- Quick Examples
- Try it now online
- Sites Powered by web2py

<http://www.web2py.com/>

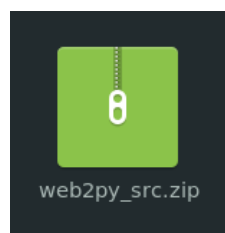
Estando en la página, daremos clic en “Download Now “

Download Now

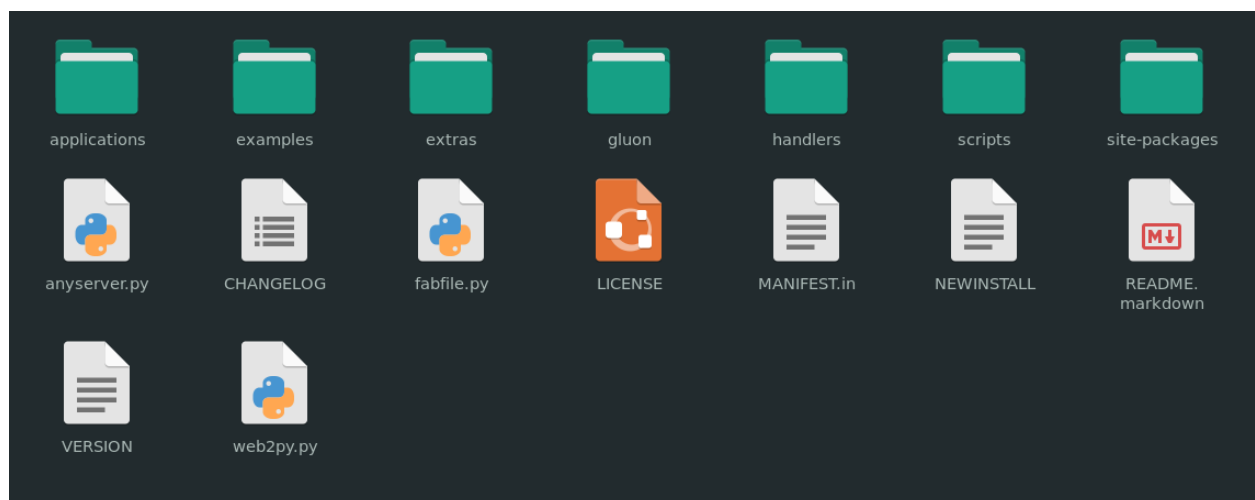
En la nueva página, nos mostrará la siguientes opciones, de la cual escogeremos “Source Code”



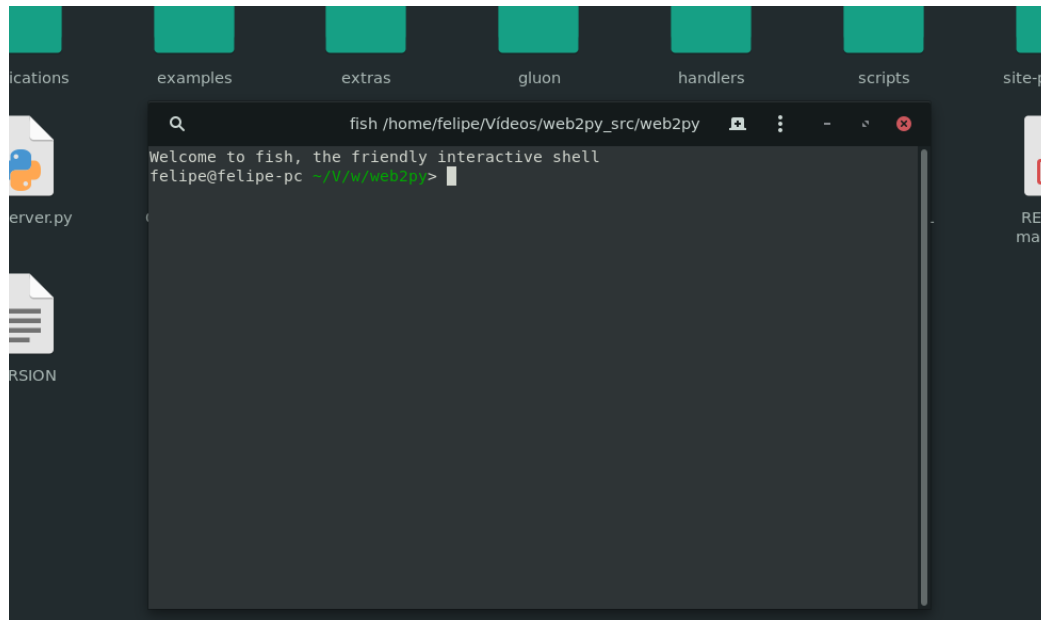
Obtendremos un archivo como el siguiente, al cual daremos clic derecho y luego en “descomprimir” o “extraer aquí”



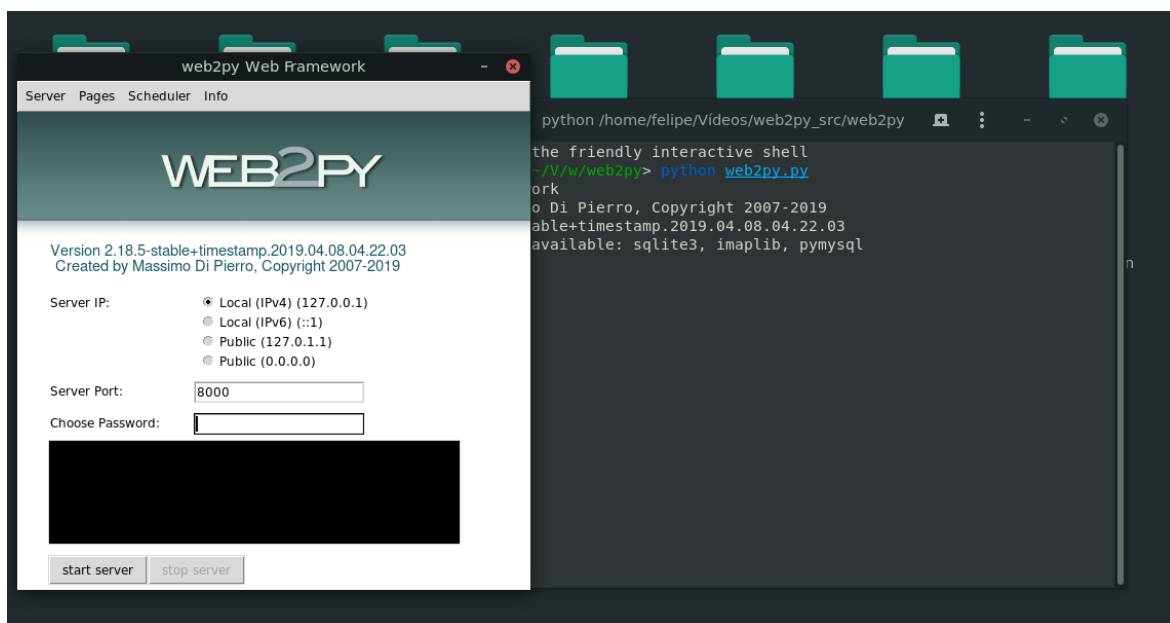
Entraremos a la carpeta:



En cualquier parte de la carpeta que no sea un archivo, daremos clic derecho y luego en “Abrir en una terminal”



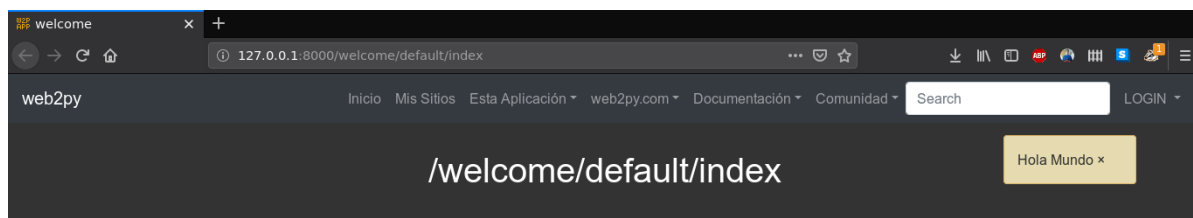
Escribiremos el siguiente comando: `python web2py.py`, al presionar enter, se desplegará la siguiente ventana



En esa ventana, se nos pide escoger una contraseña, que será la contraseña de administración, está contraseña será diferente toda vez que ingresemos en el entorno local. Después de ponerla daremos en “start server”, para iniciar el servidor.



Cuando presionemos start server, automáticamente se abrirá en el navegador predeterminado la siguiente ventana:



¡Bienvenido a web2py!

¿Cómo llegaste aquí?

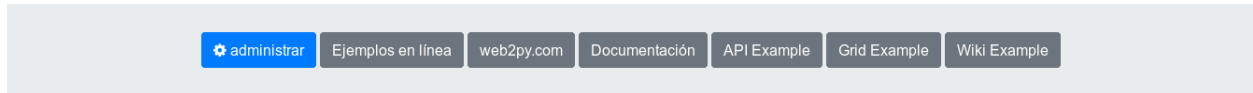
1. Usted está ejecutando web2py exitosamente
2. Usted visitó la url </welcome/default/index>
3. La cual llamó la función `index()` localizada en el archivo `web2py/applications/welcome/controllers/default.py`
4. La salida de dicha función es un diccionario que es desplegado por la vista `web2py/applications/welcome/views/default/index.html`
5. Usted puede modificar esta aplicación y adaptarla a sus necesidades

[administrar](#) [Ejemplos en línea](#) [web2py.com](#) [Documentación](#) [API Example](#) [Grid Example](#) [Wiki Example](#)

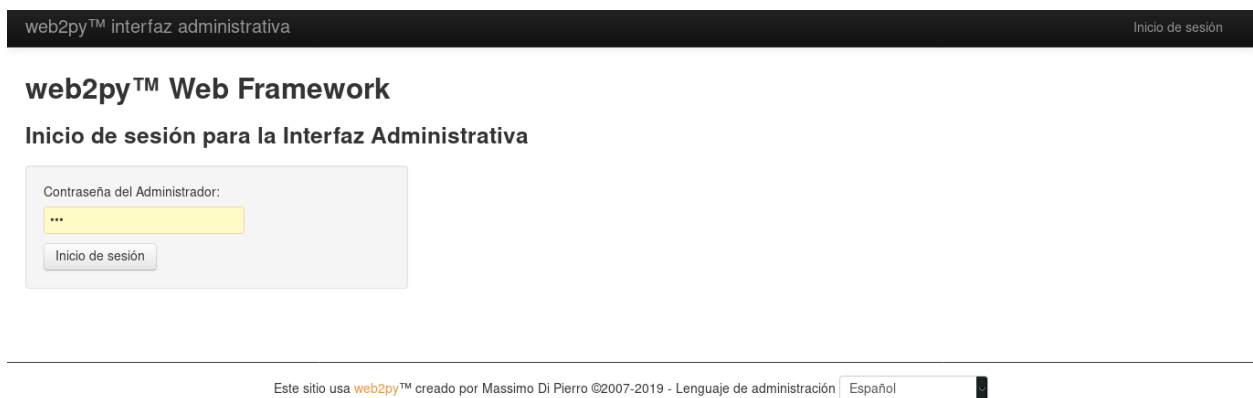
Sino se abre, podemos revisar entrando a la siguiente dirección que es la del LocalHost

127.0.0.1:8000

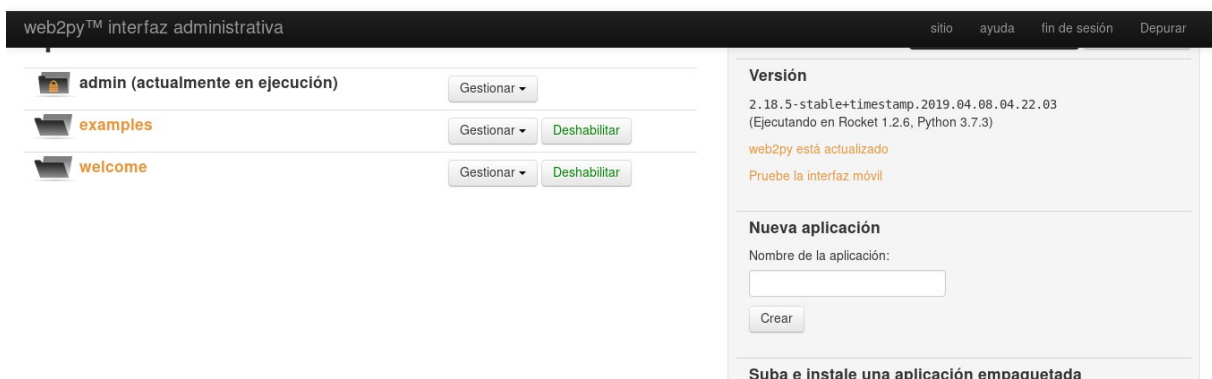
De las siguientes opciones que se muestran en el cuerpo de la página, seleccionaremos el primero: administrar



A lo cual, se nos pedirá la clave de administración que ingresamos al ejecutar el comando de inicio del framework:



Una vez iniciada la sesión, se nos mostrará:



Es momento de subir el prototipo aplicativo, que es anexo y viene empaquetado en formato .w2p

Nos desplazaremos hacia abajo en la página de administrar, yendo a las siguiente sección:

Suba e instale una aplicación empaquetada

Nombre de la aplicación:

Subir un paquete:



O obtener desde una URL:

(puede ser un repositorio git)
☐ sobreescriba la aplicación instalada

Subiremos el paquete dando clic en el icono negro de subir, y luego ajustaremos el formulario como a continuación:

Suba e instale una aplicación empaquetada

Nombre de la aplicación:

Subir un paquete:



O obtener desde una URL:

(puede ser un repositorio git)
☒ sobreescriba la aplicación instalada

El nombre welcome se da para reemplazar la aplicación de andamiaje que trae por defecto web2py. Ahora daremos clic en Instalar y esperaremos.

Si volvemos nuevamente al LocalHost:

OinkSoft Search INICIAR SESIÓN ▾

Iniciar sesion

Correo electrónico:

Contraseña:

Recuérdame (durante 30 días): ☐

[Lost your password?](#)

Desarrollado por [Andrés Collazos](#)

Cuando queramos entrar el sistema en el entorno local, ejecutamos `python web2.py`, luego ingresamos la clave de sesión administrativa, y ya podremos seguir accediendo.

Instrucciones de instalación en servicio externo

Nos dirigiremos a la página de PythonAnywhere:

<https://www.pythonanywhere.com>



[Dashboard](#) [Consoles](#) [Files](#) **Web**

oinksoft.pythonanywhere.com

[Add a new web app](#)

Configuration for [oinksoft.pythonanywhere.com](#)

Reload:

[Reload oinksoft.pythonanywhere.com](#)

Best before date:

We're happy to host your free website -- and keep it free -- for as long as you want to keep it running, but you'll need to log in at least once every three months and click the "Run until 3 months from today" button below. We'll send you an email a week before the site is disabled so that you don't forget to do that. [See here for more details.](#)

This site will be disabled on **Friday 28 February 2020**

[Run until 3 months from today](#)

[Paying users'](#) sites stay up forever without any need to log in to keep them running.

En la esquina superior derecha daremos clic en Log In si ya tenemos una cuenta creada, sino es así, daremos clic en Pricing & Signup. **Sí se está registrando, el nombre de usuario deberá ser oinksoft.**

[Send feedback](#) [Forums](#) [Help](#) [Blog](#) [Account](#) [Log out](#)

pythonanywhere

Dashboard [Consoles](#) [Files](#) [Web](#) [Tasks](#) [Databases](#)

Dashboard

Welcome, [oinksoft](#)

CPU Usage: 0% used - 0.00s of 100s. Resets in 17 hours, 42 minutes [More Info](#)

File storage: 7% full - 34.1 MB of your 512.0 MB quota

[Upgrade Account](#)

El segundo menú de opciones en la esquina superior derecha, daremos clic en “Web” y se abrirá:



➕ Add a new web app

You have no web apps

To create a PythonAnywhere-hosted web app, click the "Add a new web app" button to the left.

Daremos clic en "Add a new web app"

Create new web app ✕

Your web app's domain name

Your account doesn't support custom domain names, so your PythonAnywhere web app will live at `oinksoft.pythonanywhere.com`.

Want to change that? [Upgrade now!](#)

Otherwise, just click "Next" to continue.

Cancel

« Back

Next »

Copyright © 2011-2019 [PythonAnywhere LLP](#) — [Terms](#) — [Privacy & Cookies](#)

Luego en Next>>

En la siguiente ventana daremos clic en web2py y luego en Next

Select a Python Web framework

...or select "Manual configuration" if you want detailed control.

- » Django
- » **web2py**
- » Flask
- » Bottle
- » Manual configuration (including virtualenvs)

Seleccionaremos la contraseña de administrador, está no se cambiará en cada sesión, y debe ser lo suficientemente fuerte. Por eso se recomienda la combinación de minúsculas, mayúsculas, números y caracteres. Una vez hecho eso, daremos clic en Next y esperaremos.

Create new web app

Quickstart new web2py project

This will start web2py on your domain. If you already have a web2py app you want to use, you'll be able to use the web2py admin to install it. **WEB2PY**

Directory

Enter the path for a new directory to contain the web2py code

Admin Password

Confirm Password

Choose your password for the web2py admin site

(NB: this may take a couple of minutes)

[Cancel](#) [« Back](#) [Next »](#)

Copyright © 2011-2019 PythonAnywhere LLP — [Terms](#) — [Privacy & Cookies](#)

Ahora, requerimos decirle a PythonAnyWhere que versión de python estamos usando, para no tener errores por incompatibilidad de versiones. Cuando la app se termina de crear, nos sale lo siguiente:

oinksoft.pythonanywhere.com

[Add a new web app](#)

Configuration for oinksoft.pythonanywhere.com

Reload:

[Reload oinksoft.pythonanywhere.com](#)

Best before date:

We're happy to host your free website -- and keep it free -- for as long as you want to keep it running, but you'll need to log in at least once every three months and click the "Run until 3 months from today" button below. We'll send you an email a week before the site is disabled so that you don't forget to do that. [See here for more details.](#)

This site will be disabled on **Friday 28 February 2020**[Run until 3 months from today](#)[Paying users'](#) sites stay up forever without any need to log in to keep them running.

En ésta misma página, bajaremos hasta encontrar la sección code:

Code:

What your site is running.

Source code:	/home/oinksoft/web2py/	Go to directory
Working directory:	/home/oinksoft/	Go to directory
WSGI configuration file:	/var/www /oinksoft_pythonanywhere_com_wsgi.py	
Python version:	<div><div>2.7</div><div>✓ Save</div><div>Cancel</div></div>	

Cambiaremos la versión 2.7 de python que trae por defecto por la 3.7:

Code:

What your site is running.

Source code:	/home/oinksoft/web2py/	Go to directory
Working directory:	/home/oinksoft/	Go to directory
WSGI configuration file:	/var/www /oinksoft_pythonanywhere_com_wsgi.py	
Python version:	<div><div>3.7</div><div>✓ Save</div><div>Cancel</div></div>	

Sí vamos al link:

<https://oinksoft.pythonanywhere.com>

Nos encontraremos con el mismo menú inicial que obtuvimos al ejecutar por primera vez web2py. Iremos pues, al panel de administración como en las instrucciones anteriores y seguiremos los mismos pasos.

Suba e instale una aplicación empaquetada

Nombre de la aplicación:

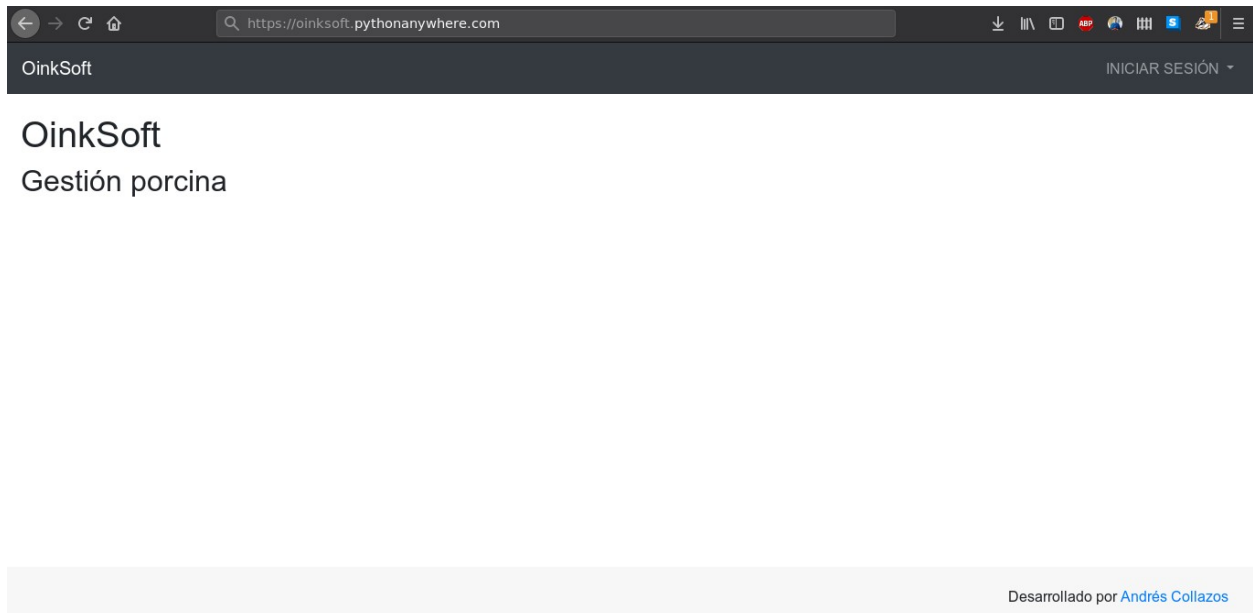
Subir un paquete:
 

O obtener desde una URL:

(puede ser un repositorio git)

☒ sobreescriba la aplicación instalada

Una vez instalado, volveremos al link de la página:



Administración de usuarios

Como se verá, el link para administrar la página ya no se encuentra disponible a la vista, así que ingresaremos al siguiente enlace:

<https://oinksoft.pythonanywhere.com/admin/default/index>

Ingresaremos la clave nuevamente si es necesario, la cual para este prototipo de aplicativo web es: `_Utp_Pass_1234_`

Una vez entramos, daremos clic en gestionar a la app welcome y luego en editar



Nos llevará a:



Donde daremos clic en “administración base de datos”

Modelos

administración base de datos

sql.log

graficación del modelo

editar



db.py definir tablas **cerda**, **lechon**, **monta**, **parto**, **semen**, **ta**

editar



menu.py

Crear

Nos mostrará todas las tablas:

OinkSoft

Bases de datos y tablas disponibles

Tables

Hooks

» db.auth_user

Registro nuevo

» db.auth_group

Registro nuevo

» db.auth_membership

Registro nuevo

» db.auth_permission

Registro nuevo

» db.auth_event

Registro nuevo

» db.auth_cas

Registro nuevo

» db.semen

Registro nuevo

» db.cerda

Registro nuevo

» db.monta

Registro nuevo

» db.tareasgestacion

Registro nuevo

» db.parto

Registro nuevo

» db.lechon

Registro nuevo

Daremos clic en Registro nuevo en la fila de db.auth_user

OinkSoft

Base de datos db tabla auth_user

Registro nuevo

Nombre:

Apellido:

Correo electrónico:

Contraseña:

Llave de registro:

Restaurar Llave de la Contraseña:

Identificador de Registro:

Enviar

Llenaremos los datos (no es necesario llenar Llave de registro para abajo), y daremos enviar.

Sí regresamos la página, veremos los usuarios granjeros que tenemos:

2 seleccionados

auth_user.id	auth_user.first_name	auth_user.last_name	auth_user.email	auth_user.password
1	Andrés	Collazos	pipecollazos1...	pbkdf2(1000,20,sha512)\$9d96d6c1a3fd8623\$13b13d782
2	Carlos Alberto	Ocamo Sepulveda	caos@utp.edu.co	pbkdf2(1000,20,sha512)\$aa23bf87f5fea5df\$02efb402daf3

Sí damos clic en el número de identificación, encontraremos un formulario de actualización, dónde también tendremos la opción de eliminar.

Características de los usuarios

Se tienen los siguientes tipo de usuario:

- **Usuario Super User:** tiene todos los permisos del sistema, ingresar al panel de administración de web2py, acceso a la base de datos, a los modelos, las vistas, los controladores. Su función primordial es la de administración los usuarios.
- **Usuario Granjero:** tiene todos los permisos de leer y escribir en los registros de la base de datos que se le proporcionan mediante formularios.

Definición del desarrollo

Para el desarrollo del proyecto se utilizó el lenguaje de programación Python, y el framework web2py, el cual provee una serie de características que posibilitan el manejo de CRUD (Crear, Remover, Actualizar y Eliminar), por lo cual la implementación reduce considerablemente los tiempos, a la vez que la curva de aprendizaje no es elevada.

Para la escritura del código fuente se siguieron las recomendaciones del mismo framework, que están basadas en PEP8, el cual es un compendio de buenas prácticas de estilo para programar en el lenguaje Python. Web2py no sigue al pie de la letra todas estas normas, sin embargo conserva gran parte de ellas. Siendo en conclusiones, las siguientes las acatadas:

- “Los ayudantes HTML se notan con mayúsculas para toda la palabra como se describió anteriormente (por ejemplo DIV, A, FORM, URL).” (Di Pierro, 2019).
- “El objeto de traducción T se escribe con mayúscula aunque se trate de una instancia de la clase, no la clase en sí. Lógicamente, el objeto traductor realiza una acción similar a la de los ayudantes de HTML; este objeto opera en parte de la conversión (*rendering*) de la información presentada. Además, T debe ser fácil de identificar en el código fuente y debe tener un nombre corto.” (Di Pierro, 2019).
- “Las clases de la DAL siguen el estándar de estilo de Python (mayúscula en la primera letra), por ejemplo Table, Field, Query, Row, Rows, etc.” (Di Pierro, 2019).

Definición de la implementación: herramientas y librerías

Para la implementación del proyecto se usaron diferentes herramientas que nos ayudan a optimizar dicho proceso.

8.1. Editor de textos

Se hizo uso del editor de textos que viene incorporado con el framework web2py

8.2. Framework Web2py

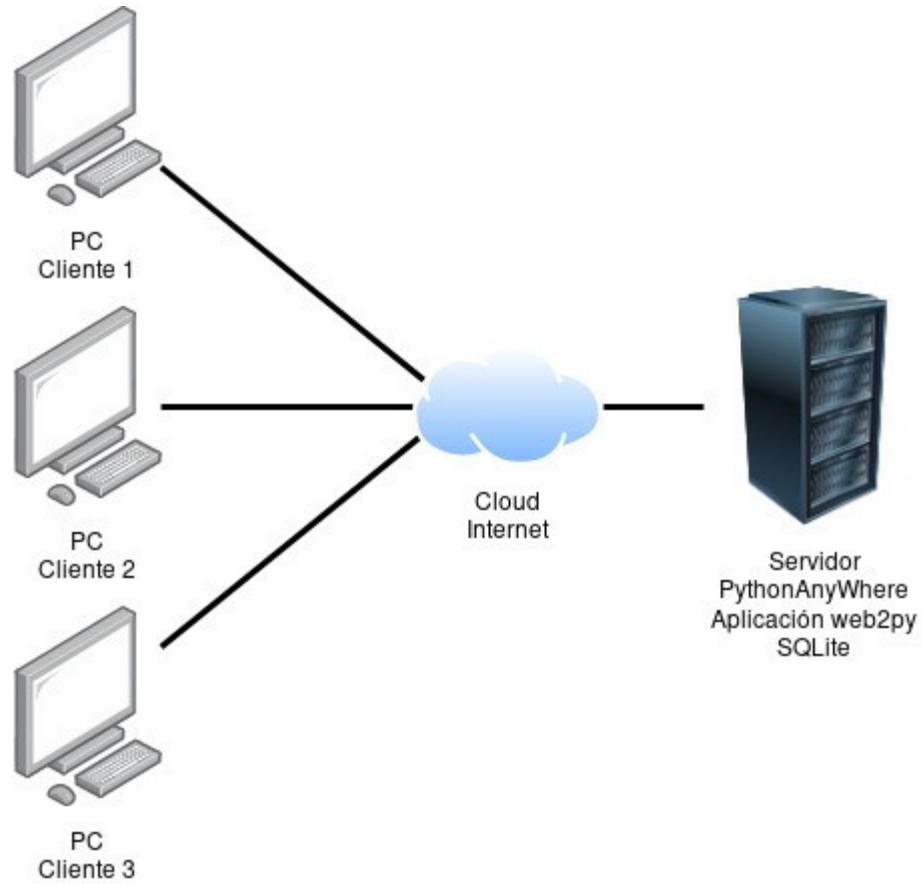
La versión que usamos del framework es la 2.18.5, esta viene como código fuente, por lo que no requiere ningún tipo de instalación. Este framework permite avanzar rápidamente en proyectos pensados en el manejo de registros, debido a que la mayoría de operaciones sobre ellos son del tipo CRUD, y web2py tiene implementados varias funcionalidades que pueden ser implementadas con facilidad. Web2py es un framework que sigue el patrón Modelo-Vista-Controlador, y en su sección de administración y edición provee una visualización estructurada en este sentido, que permite una más fácil comprensión. Se destacan lo siguiente:

- **DAL:** es un objeto que representa la conexión a la base de datos, así mismo contiene métodos para el manejo de las tablas.
- **Auth:** librería permite el manejo de usuarios, sesiones, permisos y autenticación. Siendo tan práctico, que no se necesitan más de 10 líneas de código para manejar esta importante cuestión.
- **SQLFORM:** es un método que permite el manejo de formularios
- **SQLFORM.grid:** es un método que entrega operaciones CRUD básicas para un tabla, lo cual ahorra innumerable trabajo.

- **SQLFORM.smartgrid:** es un método que se basa en SQLFORM.grid pero que permite mostrar las relaciones que contienen las tablas entre sí, a la vez que implementa las operaciones CRUD básicas.
- **SQLFORM.factory:** es un método que permite construir un formulario sin necesidad de basarse en una tabla existente en la base de datos, lo cual permite el manejo de operaciones de validación diferentes a las básicas que contiene el framework

Vista de implementación o vista física

La aplicación por ser un servicio web cuenta con la siguiente vista de implementación:



Glosario

A continuación se presentan las palabras que puedan ser un poco desconocidas, con su respectiva definición:

- **Aplicación web:** son todas las herramientas a las que se pueden acceder y utilizar para algún específico por medio de un navegador de internet.
- **Aplicación de escritorio:** son todas las herramientas que el usuario debe instalar en su computador y que funcionan con o sin internet.
- **Clase:** es la representación de las entidades o de conceptos. Una clase define un conjunto de variables que representan el estado de la cosa, y los métodos que representan el comportamiento de la cosa.
- **Despliegue:** es el proceso mediante el cual una aplicación pasa a estar lista para ser usada por los clientes destinados.
- **Editor de texto:** es una herramienta que puede ser web o de escritorio, que permite la creación y modificación de archivos de texto plano, es decir, que no tienen un formato.
- **Entorno de desarrollo:** es una aplicación que le brinda al usuario una cantidad de herramientas que le dan facilidades a la hora de desarrollar. Comúnmente están compuestos por editores de textos, tienen funciones de auto completado de palabras, compiladores, entre otros.
- **Framework:** es un esqueleto o marco que se utiliza para el desarrollo de alguna aplicación, permitiendo ahorrar tiempos al programador, al no tener éste que definir una estructural global del sistema.

- **Librería:** conjunto de funcionalidades implementadas que ofrecen una interfaz que puede ser llamada por el usuario para su aprovechamiento.
- **Modelo:** es la forma en la que se representan los datos
- **Motor de base de datos:** es una herramienta que permite el almacenamiento y manipulación de datos en forma estructurada.
- **Servidor Web:** es un software que se encarga de recibir y responder solicitudes de usuarios brindándoles respuestas

Referencias

- Di Pierro, M. (2019). Web2py Manual de Referencia Completo, 5a Edición. Recuperado de <http://www.web2py.com/books/default/chapter/41/00/prologo>