

Fernanda Fernández Ocampo

201062939

Yessenia Montiel

200707085

Miguel Ángel González

201136785

Lenguajes de Programación

Tarea Programada II

**Mantenimiento de recetas para el
restaurante francés del chef Giovanni**



Contenido

Descripción del problema	3
Diseño del Programa	4
Librerías Usadas	5
Análisis de resultados	5
Manual de usuario	6
Conclusión Personal	9



Descripción del problema

Se solicita desarrollar un sistema de consulta de recetas para el restaurante francés de Giovanni, que funcione en el sistema operativo de Linux. Se requiere el sistema, ya que el restaurante esta presentado muchos problemas con las recetas, debido que estos no cuenta con un mecanismo eficiente para obtener información de las mismas.

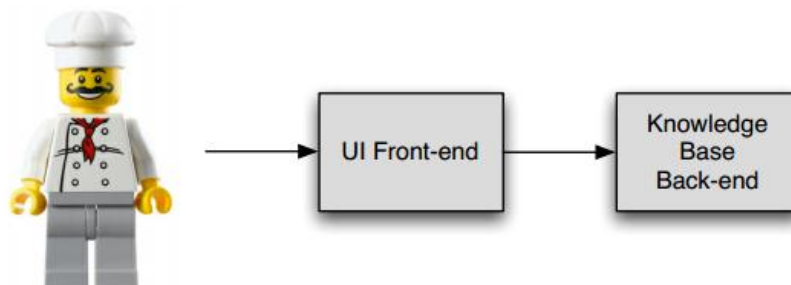
El sistema a desarrollar se divide en dos módulos. El primer módulo es el back-end, el cual debe desarrollarse en lenguaje Prolog y contiene la base de conocimientos que es una serie de declaraciones en Prolog que define todos los atributos de las recetas(nombre, ingredientes, pasos, autor, origen, y su temperatura).

A su vez, el back-end será el encargado de manejar la base de conocimientos de las recetas, y de responder las consultas hechas por el usuario.

El segundo módulo es el front-end, es una interfaz gráfica desarrollada en un lenguaje alternativo. El front-end está conformado por dos modalidades:

- mantenimiento y
- consulta de los datos de los diferentes recetas.

En la modalidad de mantenimiento de datos, les permite a los encargados ingresar nuevos datos de las recetas.



Las consultas le permite a los encargados ingresar las características de las recetas que desea consultar en el Front-end, y cuando el usuario haya terminado de digitarlas, el Front-end deberá comunicarse con el Back-end, para que realice las consultas pertinentes a la base de conocimientos, y devolver la información al Front-end, para que sea desplegada al usuario correctamente.

Diseño del Programa

El programa fue diseñado de la siguiente manera, **(ver figura más adelante).**

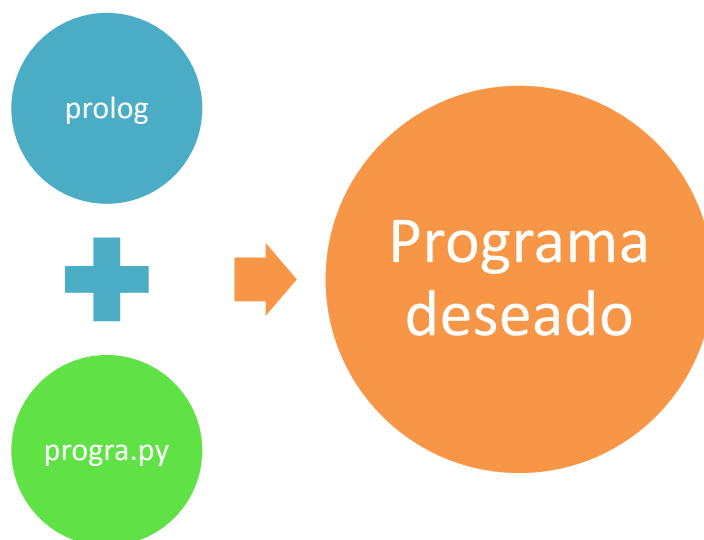
Escogencia de lenguaje y librería.

Se escogió en conjunto el lenguaje Python para la implementación del back-end, porque ya se tiene experiencia programando la interfaz gráfica y además es un lenguaje muy permisivo y simple de usar además realizando la investigación se encontró una librería llamada pyswip el cual utiliza una sintaxis muy similar a la utilizada por prolog lo cual simplificaría las consultas y manejo de la base de conocimiento.

Solución.

Se crearon archivos de python para resolver esta tarea programada. Este contiene las funciones que se encargan de inicializar, agregar y consultar datos en la base de conocimientos de prolog, también muestra la interfaz gráfica y presentarle al usuario las funcionalidades de modo consulta y mantenimiento.

Luego de la recolección de los datos requeridos este archivo utiliza el ejecutar el proceso deseado.



Librerías Usadas

Para el desarrollo de dicho programa se utilizaron diferentes librerías las cuales son:

- **PYSWIP:** es un puente entre Python y SWI-Prolog que permite realizar consultas de los predicados en SWI-Prolog desde Python.
- **Tkinter:** Es una interfaz gráfica de GUI, basada en las librerías gráficas TCL/TK, interface "de-facto" preinstalada con python, es la generalmente recomendada para proyectos triviales y/o de aprendizaje. Además una característica de Tkinter es que es portable, lo que es muy favorable para el desarrollo del sistema.

from Tkinter import *	Se utiliza para realizar la interfaz del programa.
Import tkMessageBox	Se utiliza para enviar ventanas simples con mensajes para el usuario.
Import tkSimpleDialog	Se utiliza para hacer preguntas simples con mensajes al usuario.
From string import *	Se utilice para el manejo de strings.
Import os	Se utiliza para abrir el manual de usuario.
Import sys	Para salir del programa y cerrar la ventana principal.
From pyswip import *	Se utiliza para realizar la conexión de python-prolog.

Análisis de resultados

Se cumplieron con los objetivos establecidos en el documento de la tarea.

- Registrar nuevas recetas a la base de conocimientos.
- Consultar información de las recetas.

Manual de usuario

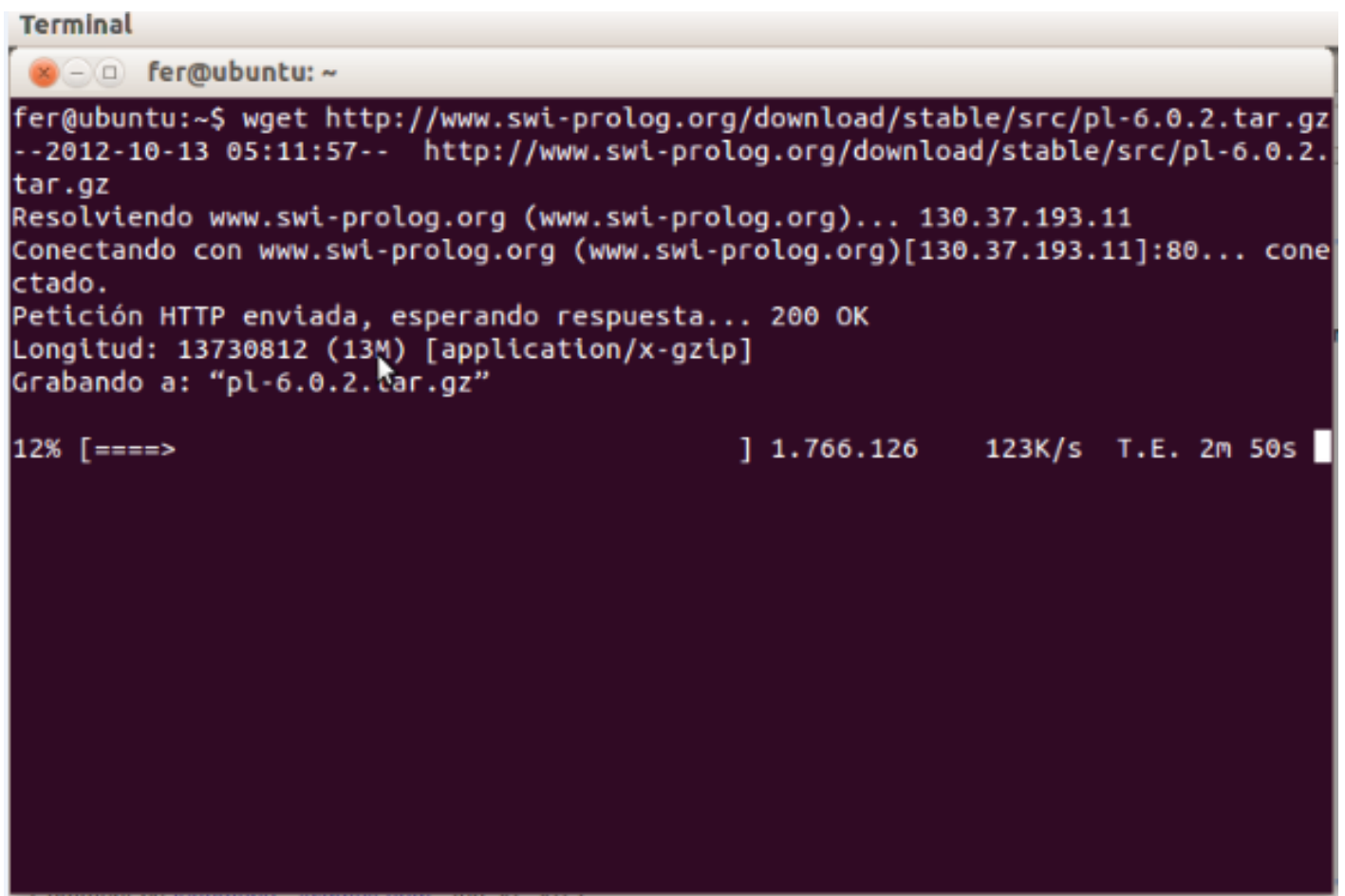
- 1) Primeramente debes ingresar a la página <http://code.google.com/p/pyswip/wiki/INSTALL> para seguir con los pasos descritos al final de la misma.

Here's what I did as of March 5th, 2012 to get this to work (in easy copy-and-paste format, for those brave enough to trust me ;P):

```
wget http://www.swi-prolog.org/download/stable/src/pl-6.0.2.tar.gz
tar xzvf pl-6.0.2.tar.gz
cd pl-6.0.2/
./configure --prefix=/usr --enable-shared
make && sudo make install
cd packages/clpqr/
./configure --prefix=/usr
make && sudo make install
sudo ln -s /usr/lib/swipl-6.0.2/lib/i686-linux/libswipl.so /usr/lib/libpl.so
sudo ln -s /usr/lib/swipl-6.0.2/lib/i686-linux/libswipl.so.6.0.2 /usr/lib/.
wget http://pyswip.googlecode.com/files/pyswip-0.2.2.zip
unzip pyswip-0.2.2.zip
cd pyswip-0.2.2/
sudo python setup.py install
```

Then fire up the Python shell and try out the example above and hopefully everything works! :D

- 2) Abrimos la terminal de Ubutu y empezamos a desarrollar cada uno de los puntos anteriores.



```
Terminal
fer@ubuntu: ~
fer@ubuntu:~$ wget http://www.swi-prolog.org/download/stable/src/pl-6.0.2.tar.gz
--2012-10-13 05:11:57-- http://www.swi-prolog.org/download/stable/src/pl-6.0.2.
tar.gz
Resolviendo www.swi-prolog.org (www.swi-prolog.org)... 130.37.193.11
Conectando con www.swi-prolog.org (www.swi-prolog.org)[130.37.193.11]:80... cone
ctado.
Petición HTTP enviada, esperando respuesta... 200 OK
Longitud: 13730812 (13M) [application/x-gzip]
Grabando a: "pl-6.0.2.tar.gz"

12% [====> ] 1.766.126 123K/s T.E. 2m 50s
```

- 3) Insertamos las librerías antes descritas para la perfecta ejecución del programa desarrollado en python

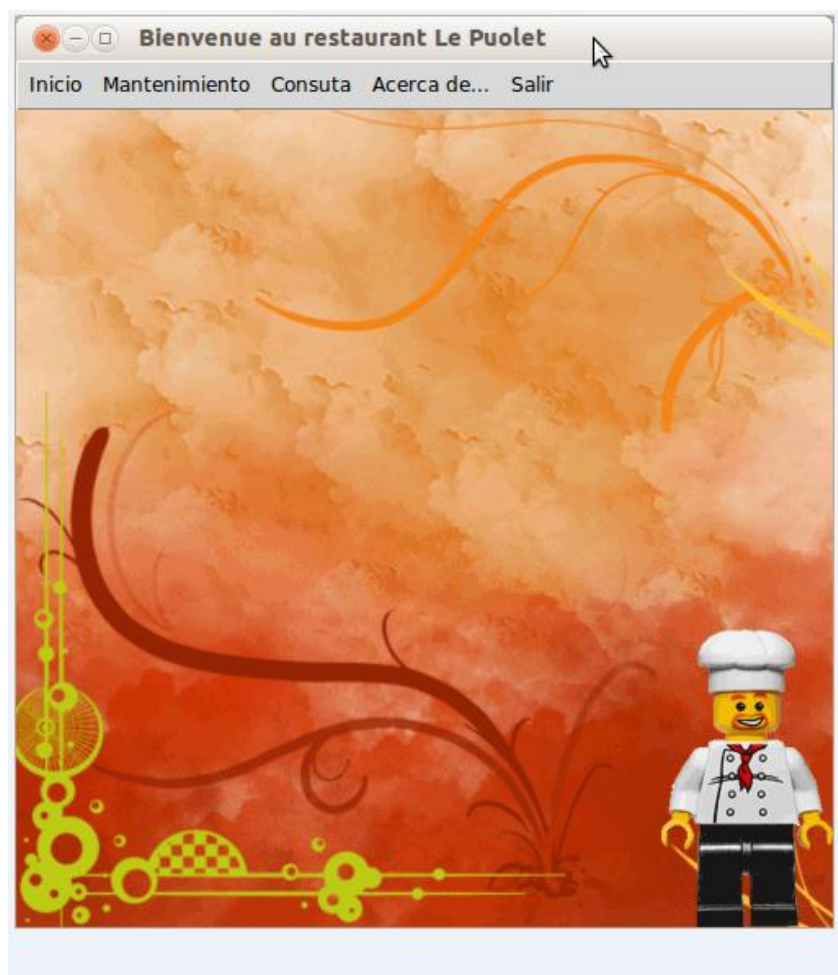
```
# -*- coding: utf-8 -*-  
#Lenguajes de Programacion  
#Administracion de tecnologias de informacion  
#Segunda tarea programada  
  
#Fernanda Fernandez-Miguel Angel Gutierrez-Yessenia  
  
##### Importación de Librerías  
  
from Tkinter import * #importar libreria gráfica Tk  
import tkMessageBox #importar el módulo para los mensajes  
import os #importar la clase os, que ayuda a cerrar el programa  
from pyswip import Functor, Variable, Query, call,   
import tkSimpleDialog  
from string import * #Modulo de manejo de Strings  
import sys #Modulo sys para cerrar el programa  
from pyswip import * #Importa la libreria pyswip  
  
#Inicia prolog  
p = Prolog()
```

- 4) Para ejecutar el programa ejecute desde la terminal el siguiente comando

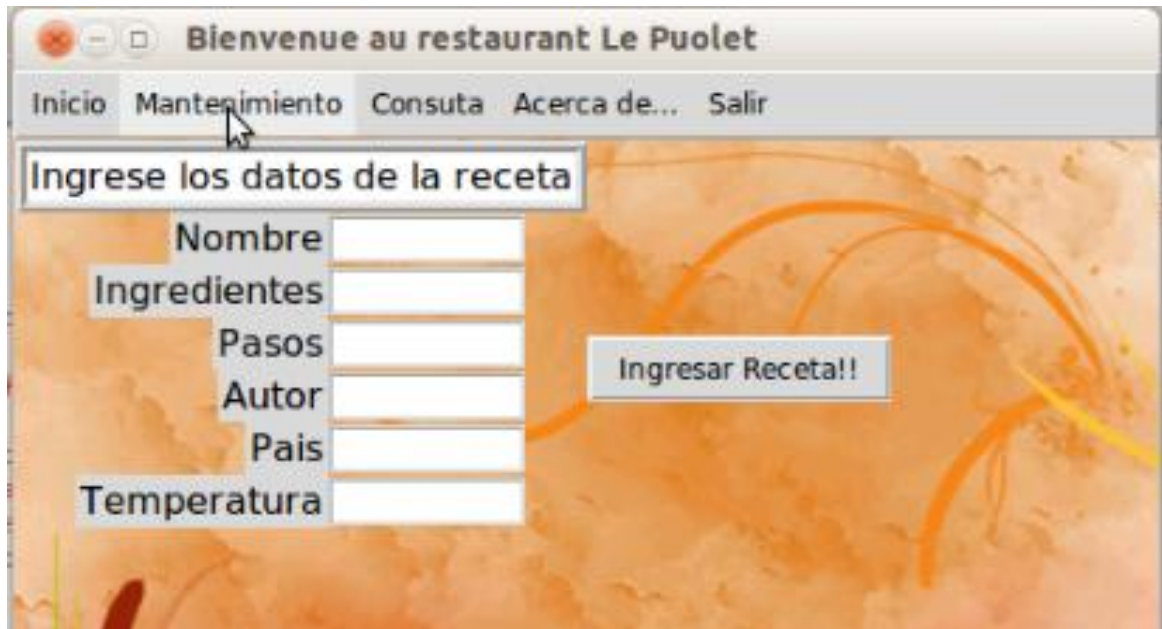
- Colóquese en el directorio principal y ejecute:
- python progra.py

En caso de dar error, abra el IDLE de python 2.7 y ejecute el programa desde allí.

Al terminar la instalación y ejecutar el programa principal se abre la siguiente ventana.

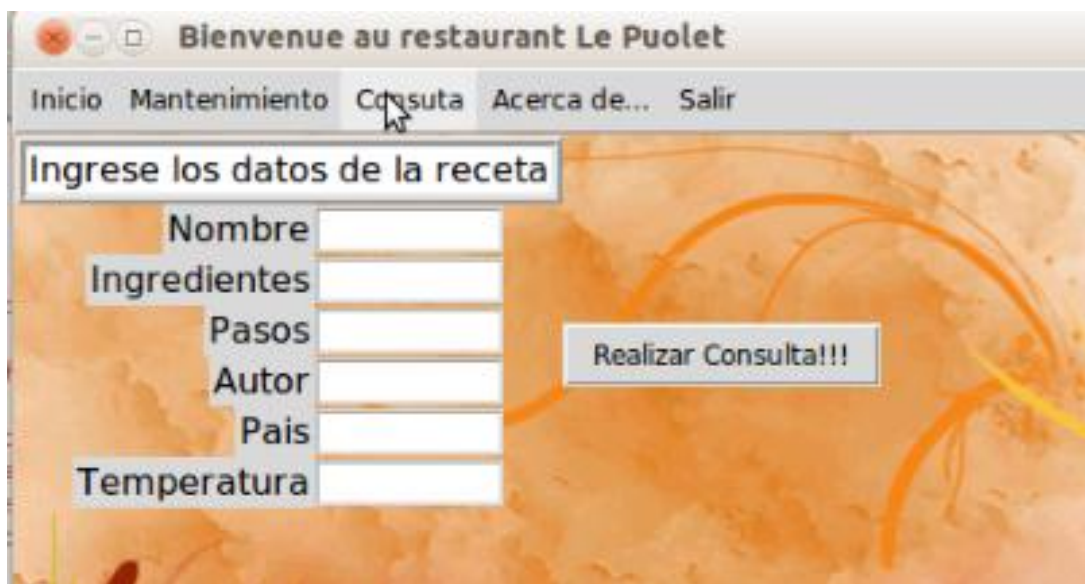


Luego se le abrirá una ventana en la cual podrá hacer consultas e ingresar información de las recetas. A continuación se presenta el caso cuando le dan click en el botón “Mantenimiento”, donde podrá ingresar los datos de las recetas



Sí la receta fue ingresado a la base de conocimientos correctamente se abrirá la ventana con el mensaje de éxito, de lo contrario el mensaje será de error.

Ventana para realizar consulta, la cual si se dejan todos los campos vacíos, el resultado son todas las recetas ingresadas a la base de conocimiento, en cambio sí se ingresa solo un dato, el resultado será todas las recetas que en su información posean ese o la combinación de los datos.



Conclusión Personal

Como conclusión del grupo consideramos importante conocer las técnicas y las librerías necesarias para unir dos lenguajes de programación, ya que para desarrollar sistemas de gran tamaño se debe aprovechar las ventajas y herramientas que nos ofrecen diferentes lenguajes.

Se considera que la tarea está adaptada a los conocimientos proporcionados por el curso, como lo es crear una base de conocimientos y realizarle consultas.

Los integrantes del grupo trabajaron igualmente en todo el proceso de desarrollo de esta tarea. No se usó mucho la herramienta Github ya que nos reuníamos de forma presencial y se utilizó una única versión en un computador para facilitar el orden.

