



UNIVERSIDAD NACIONAL  
DEL LITORAL

# Tecnicatura Universitaria en Software Libre

## Trabajo Final

### Proyecto Gestión Cooperativa con Tryton ERP

**Lugar de Implementación: Cooperativa de  
Trabajo COOPEVIC Ltda.**



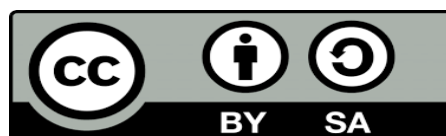
**Autor:** Hernán Albornoz

### **¡Copia este texto!**

Los textos que componen este trabajo se publican bajo formas de licenciamiento que permiten la copia, la redistribución y la realización de obras derivadas, siempre y cuando éstas se distribuyan bajo las mismas licencias libres y se cite la fuente. El copyright de los textos individuales corresponde a los respectivos autores.

Este trabajo está licenciado bajo un esquema Creative Commons Atribución Compartir Igual (CC-BY-SA) 4.0 Internacional.

<http://creativecommons.org/licenses/by-sa/4.0/deed.es>



## Datos del autor

La autoría de este proyecto corresponde a Carlos Hernán Albornoz, egresado de la Facultad de Ciencia y Tecnología de la Universidad Autónoma de Entre Ríos, con los títulos de Analista Programador y Analista Superior en Sistemas. Es además miembro del Grupo de Usuarios de Software Libre de Victoria Entre Ríos y de la Comunidad de usuarios de Huayra GNU/Linux, como así también, socio y fundador de la Cooperativa de Trabajo de Software Libre Coopevic Ltda. y a partir del año 2017 Coordinador y organizador del Festival Latinoamericano de Instalación de Software Libre FLISoL Victoria. Se desempeña además como capacitador en TIC – Tecnología de la Información y la Comunicación y en el Taller de Asistente en Armado y Mantenimiento de Computadoras en el Centro Comunitario N° 30 dependiente de la Dirección de Educación de Jóvenes y Adultos del Consejo General de Educación de Entre Ríos. La elección de inscribirse a la Tecnicatura Universitaria en Software libre de la UNL, fue principalmente por ser un entusiasta, activista, militante y defensor del software libre.

## Agradecimientos

Para la realización de este proyecto se contó con la colaboración y ayuda de las siguientes personas:

- Ing. Gustavo Courault en la dirección y asesoramiento.
- Bioingenieros Ingrid Spessotti, Carlos Scotta, Francisco Moyano Casco, Fernando Sassetti y Francisco Arata que juntos trabajan en la implementación de GNU Health en diversos hospitales públicos de la ciudad de Paraná y Diamante Entre Ríos, ellos me brindaron tutoriales y material de consulta de instalación
- Luciano Rossi socio de la Cooperativa de Software Libre Gcoop que también me guió en la instalación de Tryton a través del correo electrónico.
- A los Analistas de Sistemas e informáticos, Carlos Burgos, José Ferrari, Gabriel Balbi, Matías Carrizo y Helvio Bruera, que son compañeros, amigos y socios de la Cooperativa de Software Libre COOPEVIC Ltda. Por el apoyo desde un primer momento.
- A Fabián Delgado, coterraneo, amigo y compañero durante toda la cursada de la tecnicatura de Software Libre y en donde compartimos varias horas de estudio.
- A todos aquellos amigos de la Comunidad del Software Libre que de alguna u otra manera también me guiaron o alentaron en este proyecto.
- Y por último a mi señora y a mi hijo Valentín que me aguantaron en vacaciones para que yo pudiera culminar con este trabajo final.

## Indice

<u>Fundamentos del proyecto</u>	<u>Pág. 8</u>
<u>Importancia del proyecto</u>	<u>Pág. 8</u>
<u>Manejo de actas:</u>	<u>Pág. 8</u>
<u>Manejos de Socio:</u>	<u>Pág. 8</u>
<u>Destinatario del proyecto</u>	<u>Pág. 9</u>
<u>Cooperativa de Trabajo COOPEVIC Ltda.</u>	<u>Pág. 9</u>
<u>Desarrollo de Software:</u>	<u>Pág. 9</u>
<u>Diseño de Redes Cableadas e Inalambricas:</u>	<u>Pág. 9</u>
<u>Servicio Tecnico de PC:</u>	<u>Pág. 9</u>
<u>Capacitación:</u>	<u>Pág. 10</u>
<u>Problemas que resuelve este proyecto</u>	<u>Pág. 10</u>
<u>Aplicación de conceptos técnicos aprendidos en el trayecto</u>	<u>Pág. 10</u>
<u>Licencias involucradas en el proyecto</u>	<u>Pág. 11</u>
<u>Documentación</u>	<u>Pág. 11</u>
<u>Análisis del estado del arte</u>	<u>Pág. 11</u>
<u>Objetivo General</u>	<u>Pág. 13</u>
<u>Objetivos Específicos</u>	<u>Pág. 14</u>
<u>Metodología</u>	<u>Pág. 14</u>
<u>Guía ágil para instalación de Tryton con localización Argentina</u>	<u>Pág. 16</u>
<u>Instalación de Tryton 4.0 sobre Debian 8 Jessie</u>	<u>Pág. 16</u>
<u>Preparativos</u>	<u>Pág. 16</u>
<u>Instalación de pip y virtualenvwrapper</u>	<u>Pág. 19</u>
<u>pip</u>	<u>Pág. 19</u>
<u>virtualenv</u>	<u>Pág. 19</u>
<u>virtualenvwrapper</u>	<u>Pág. 19</u>
<u>Configuración de virtualenvwrapper</u>	<u>Pág. 19</u>
<u>Creación del entorno e instalación de Tryton</u>	<u>Pág. 22</u>
<u>Instalación de módulos del repositorio oficial de Tryton</u>	<u>Pág. 23</u>

<a href="#">El archivo de configuración trytond.conf</a>	<a href="#">Pág. 28</a>
<a href="#">La contraseña de super usuario</a>	<a href="#">Pág. 29</a>
<a href="#">Creación del usuario Postgresql cuya clave será “tryton”</a>	<a href="#">Pág. 30</a>
<a href="#">Crear base de datos</a>	<a href="#">Pág. 31</a>
<a href="#">Instalación las dependencias:</a>	<a href="#">Pág. 33</a>
<a href="#">Instalación de git para poder descargar los módulos de github</a>	<a href="#">Pág. 33</a>
<a href="#">Instalación de las dependencias en el entorno virtual</a>	<a href="#">Pág. 33</a>
<a href="#">Descarga e instalación de requerimientos de pyafipws</a>	<a href="#">Pág. 34</a>
<a href="#">Instalación en el directorio raíz de httplib2 y M2Crypto</a>	<a href="#">Pág. 34</a>
<a href="#">Copiar M2Crypto* a site package</a>	<a href="#">Pág. 35</a>
<a href="#">Cambiar al directorio site-packages del virtualenv</a>	<a href="#">Pág. 35</a>
<a href="#">Antes de instalar Pillow, asegurarse de instalar las dependencias:</a>	<a href="#">Pág. 35</a>
<a href="#">Instalación de Pillow</a>	<a href="#">Pág. 36</a>
<a href="#">Descarga e instalación de las dependencias PySimpleSoap y suds:</a>	<a href="#">Pág. 36</a>
<a href="#">Instalar módulo pyafipws</a>	<a href="#">Pág. 37</a>
<a href="#">Otra forma de descargar e instalar pysimplesoap</a>	<a href="#">Pág. 37</a>
<a href="#">Cambiano al directorio de módulos de tryton</a>	<a href="#">Pág. 37</a>
<a href="#">Instalación de los módulos de trytonar y de cooperativas de Gcoop</a>	<a href="#">Pág. 38</a>
<a href="#">Clonación de los módulo de Tryton Argentina y de Gcoop</a>	<a href="#">Pág. 38</a>
<a href="#">Renombrar las siguientes carpetas de módulos</a>	<a href="#">Pág. 38</a>
<a href="#">Cambiar a la rama 4.0 e instalar los módulos de python</a>	<a href="#">Pág. 38</a>
<a href="#">Otra forma de Instalar los módulos, es usando un script</a>	<a href="#">Pág. 39</a>
<a href="#">Creamos el script install-trytonar.sh</a>	<a href="#">Pág. 40</a>
<a href="#">Script install-trytonar.sh</a>	<a href="#">Pág. 40</a>
<a href="#">inicializar la base de datos</a>	<a href="#">Pág. 41</a>
<a href="#">Arrancamos el servidor!</a>	<a href="#">Pág. 41</a>
<a href="#">Descargar el cliente tryton desde la página oficial</a>	<a href="#">Pág. 42</a>
<a href="#">Si lo dejamos en la carpeta descargas</a>	<a href="#">Pág. 42</a>
<a href="#">Habilitar los permisos de ejecución</a>	<a href="#">Pág. 42</a>
<a href="#">Inicialización de Tryton</a>	<a href="#">Pág. 43</a>

<a href="#"><u>Editamos los perfiles</u></a>	<a href="#"><u>Pág. 43</u></a>
<a href="#"><u>Creamos la base de datos</u></a>	<a href="#"><u>Pág. 44</u></a>
<a href="#"><u>Cargamos los datos del usuario</u></a>	<a href="#"><u>Pág. 44</u></a>
<a href="#"><u>Inicialización de Tryton</u></a>	<a href="#"><u>Pág. 45</u></a>
<a href="#"><u>Configuración de módulos y usuarios</u></a>	<a href="#"><u>Pág. 46</u></a>
<a href="#"><u>Actualización del sistema</u></a>	<a href="#"><u>Pág. 46</u></a>
<a href="#"><u>Administración de Módulos</u></a>	<a href="#"><u>Pág. 47</u></a>
<a href="#"><u>Módulo Gestión Cooperativa</u></a>	<a href="#"><u>Pág. 47</u></a>
<a href="#"><u>Socios</u></a>	<a href="#"><u>Pág. 47</u></a>
<a href="#"><u>Vacaciones</u></a>	<a href="#"><u>Pág. 48</u></a>
<a href="#"><u>Sanciones</u></a>	<a href="#"><u>Pág. 50</u></a>
<a href="#"><u>Reuniones</u></a>	<a href="#"><u>Pág. 50</u></a>
<a href="#"><u>Recibos</u></a>	<a href="#"><u>Pág. 51</u></a>
<a href="#"><u>Balance Social Cooperativo</u></a>	<a href="#"><u>Pág. 53</u></a>
<a href="#"><u>Publicación en gitlab</u></a>	<a href="#"><u>Pág. 58</u></a>
<a href="#"><u>Conclusión</u></a>	<a href="#"><u>Pág. 59</u></a>
<a href="#"><u>Bibliografía</u></a>	<a href="#"><u>Pág. 62</u></a>

## Fundamentos del proyecto

El siguiente proyecto se creó principalmente para la finalización de la carrera de Tecnicatura Universitaria en Software Libre perteneciente a la Universidad Nacional del Litoral, pero también para ser implementado en la Cooperativa de Software Libre Coopevic Ltda. En la cual soy socio fundador.

Dicho proyecto pretende mejorar el trabajo diario que se realiza en la cooperativa y agilizar el trabajo administrativo de la misma.

## Importancia del proyecto

Este proyecto es importante, porque está desarrollado con una serie de módulos que permiten llevar la gestión de una Cooperativa. Esto supone:

**Manejo de actas:** permite la carga de las reuniones de la cooperativa (ya sean reuniones de consejo, asamblea ordinaria o extraordinaria) señalando qué socios participaron de las mismas.

**Manejos de Socio:** permite llevar un legajo digital del socio con todos los datos legales necesarios y otros datos útiles como vacaciones, participación en reuniones y sanciones aplicadas. Además el sistema tiene un completo workflow que permite generar los Recibos de adelanto de excedentes de los socios de cooperativas de trabajo (y la imputación contable de los mismos).

Se desarrolló además la posibilidad de exportar un "Balance Social Cooperativo" con toda la información que la cooperativa impute a las cuentas analíticas del Balance Social Cooperativo y todas las Notas que se realicen para



tener una memoria narrativa de las diversas actividades realizadas. En este informe el sistema muestra además un resumen de los socios, las altas y bajas y la participación en reuniones. Este Balance Social Cooperativo permite cuantificar y visibilizar actividades económicas y sociales que de otra forma quedan ocultas en la gestión diaria de la empresa cooperativa.

## **Destinatario del proyecto**

---

Este proyecto será implementado en la Cooperativa de Trabajo COOPEVIC Ltda. Ubicada en la ciudad de Victoria Entre Ríos. Esta empresa cooperativa se dedica a implementar proyectos de tecnología basados en Software Libre.

## **Cooperativa de Trabajo COOPEVIC Ltda.**

---

**Desarrollo de Software:** Desarrollo de software de gestión con herramientas de última generación.

**Diseño de Redes Cableadas e Inalambricas:** Diseño y el armado de redes corporativas y gubernamentales, enlaces Inalámbricos y soluciones de conectividad.

**Servicio Técnico de PC:** Soporte técnico de equipos de computación.

**Capacitación:** Cursos de capacitación a empresas y a equipos de trabajo.

La misma cuenta con el siguiente equipamiento:

6 PCs nuevas con 4 GB RAM y 500 GB de capacidad de disco duro y SO Debian 9

2 PCs con 2 GB RAM y 160 GB de capacidad de disco duro viejas con Xubuntu

1 Servidor con 4 GB RAM y 500 GB de capacidad de disco duro con Debian 8

Este proyecto fue informado en reunión de comisión en la cooperativa y se

decidió por unanimidad implementar este módulo de Gestión Cooperativa para mejorar el trabajo realizado a diario en la misma.

## Problemas que resuelve este proyecto

Pensando en la necesidad de nuestra cooperativa en cuanto a los movimientos internos, es que decidí implementar este sistema de gestión para facilitar el manejo diario de la información administrativa y contable de las mismas adaptado a la legislación argentina. Esto mejoraría mucho el manejo de la cooperativa, ya que a la fecha, todo se hace por escrito y no se tiene por lo tanto un registro digitalizado.

## Aplicación de conceptos técnicos aprendidos en el trayecto

Con esta implementación, se aplican algunos conocimientos técnicos aprendidos en el trayecto de administración, como por ejemplo: - Uso de la terminal o consola como usuario root o administrador para realizar la instalación del Servidor y cliente de Tryton, como así también, la instalación del motor de base de datos PostgreSQL. - Se aplica también algunos conocimientos sobre redes, al instalar el servidor y cliente de Tryton y los módulos de cooperativas para poder ser utilizado por todas las PCs que conforman la red. - También aplicamos algunos conocimientos sobre base de datos al instalar y utilizar el motor de base de datos PostgreSQL.

## Licencias involucradas en el proyecto

Tryton se basa en tecnología Python/XML trabajando sobre una base de datos PostgreSQL lo que lo hace altamente escalable, seguro y customizable. Tryton se organiza en una arquitectura de tres capas: El cliente Tryton, el servidor Tryton y la Base de datos. La plataforma junto con los módulos oficiales están cobijados por la licencia GPLv3.

## Documentación

Este proyecto se publicará en formato PDF y ODT en GitLab para que lo puedan descargar y compartir. <https://gitlab.com/Hertux/Trabajo-Final>, también se publicará en un blog personal <https://www.hernanalbornoz.wordpress.com>.

## Análisis del estado del arte

La cooperativa de Software Libre GCOOP realizó una serie de desarrollos para generar un ERP para Cooperativas de la Argentina. No solo lo desarrolló sino que también, es utilizado por dicha cooperativa.

Desde abril de 2013, GCOOP, forma parte del "Cluster Software Libre, Consorcio de Tecnologías Libre y Abiertas" que con el apoyo del FONTAR del Ministerio de Ciencia y Tecnología de la Nación tiene como objetivo el desarrollo de diversos proyectos de Software Libre.

El trabajo que realizaron, en conjunto con la comunidad de Tryton Argentina, hizo posible la localización de Tryton para la Argentina. Esto supone un plan contable base y un plan contable para cooperativas (que puede ser modificado de

forma sencilla por cada empresa). Además desarrollaron el módulo que permite la Facturación Electrónica contra la AFIP, funcionalidad que se suma a la ya existente factura manual, pre impresa y con impresora fiscal. A esto se agregan diversas importaciones y exportaciones que permiten la integración con sistemas fiscales nacionales y provinciales.

Por otro lado desarrollaron una serie de módulos que permiten llevar la gestión de una Cooperativa.

Todo el trabajo fue liberado y documentado con documentación en línea y video tutoriales.

Con este trabajo y gracias a la lógica de negocios del Software Libre la tecnología ERP queda al alcance de PyMEs y Cooperativas<sup>1</sup>.

Además, la Universidad Nacional del Litoral desarrolla, en conjunto con las Cooperativas Jóvenes de San Cayetano y Trabajo por un mañana mejor de la provincia de Santa Fe, un software libre para ayudar a fortalecer los procesos de gestión de las iniciativas socio-económicas colectivas<sup>2</sup>.

El proyecto busca reconocer las potencialidades del uso del software de gestión para cooperativas de trabajo en clave de economía social y solidaria. El proceso se inicia con el reconocimiento y sistematización de los procesos de negocio de las cooperativas. A partir de la información relevada, se propone implementar un sistema informatizado de gestión basado en software libre (denominado Tryton ERP). Para ello, se realizan reuniones con los integrantes de

---

1 <https://www.gcoop.coop/gestion-cooperativa-con-tryton-erp>

2 <https://www.argentina.gob.ar/noticias/el-software-libre-como-herramienta>

las cooperativas y expertos, talleres y capacitaciones para el uso del sistema.

Todas las actividades están financiadas por el Programa de Cooperativismo y Economía Social en la Universidad, perteneciente a la Secretaría de Políticas Universitarias del Ministerio de Educación de la Nación. Desde el 2013, el programa viene promoviendo el fortalecimiento del vínculo entre las universidades y las entidades de la economía social e impulsando su desarrollo, innovación y promoción, a partir del aporte de la universidad a la producción de investigación aplicada y la generación y transferencia de conocimientos y tecnologías.

Además cuento con el aporte de material de compañeros de bioingeniería que realizaron la instalación de Tryton para el Hospital Lister de Seguí y en un centro de salud de la ciudad de Diamante para que dichos centros puedan utilizar el sistema GNU Health. Y el módulo de cooperativas se puede obtener del trabajo realizado y liberado por la cooperativa GCOOP.

## **Objetivo General**

---

- Mejorar la organización de la cooperativa, automatizar los procesos que ahora se hacen a mano y mejorar la gestión.
- Elegir el software para la implementación del proyecto.
- Estudiar el uso del software y los módulos.
- Adaptarlo a las necesidades de la gestión cooperativa.
- Realizar la instalación del software y de los módulos que sean necesarios.
- Capacitar a los socios que van a utilizar este software de Gestión

Cooperativa.

## **Objetivos Específicos**

---

- Cargar los datos útiles de los socios de la cooperativa, permitiendo tener el legajo del socio dentro del sistema, el estado, fecha de ingreso, etc.
- Cargar los días que tiene asignados de vacaciones cada asociado y luego permitir cargar un registro por cada licencia o vacaciones que el socio se tome.
- Cargar sanciones a un socio con los Tipos que figuran en el Estatuto base de una Cooperativa de Trabajo: Llamado de Atención, Apercibimiento y Exclusión con el causante y el descargo presentado por el socio.
- Cargar las Reuniones de Consejo y Asambleas que realiza la cooperativa, señalando los socios presentes y los temas tratados como área de texto o archivo adjunto (Tipos de Reunión: reunión de consejo, Asamblea Ordinaria o Extraordinaria).
- Permitir crear el Recibo de Anticipo de Retornos a Cuenta de Excedentes que los socios de las cooperativas de trabajo reciben como prestación del trabajo que realizan.
- Cargar las actividades realizadas y valorizar actividades con impacto social que de otra manera quedan invisibilizadas en un Balance Contable.

## **Metodología**

---

Para la elección del software se buscó alguna herramienta que fuera libre y

que se estuviera utilizando con cierto éxito en algunas empresas cooperativas. En esa búsqueda encontramos a Tryton, es una plataforma informática general de alto nivel, que organiza en una arquitectura de tres capas: el cliente Tryton, el servidor Tryton y la Base de datos (principalmente PostgreSQL) y la plataforma junto con los módulos oficiales están cobijados por la licencia GPLv3.1.

Luego de la elección del software, se comenzó a estudiar la documentación correspondiente a Tryton al módulo de Gestión Cooperativa y la documentación del motor de base de datos PostgreSQL.

La adaptación a las necesidades de la cooperativa llevaría muy poco tiempo ya que el módulo es específico para la gestión cooperativa.

En cuanto a la instalación del software, se ha consultado desde la página oficial de Tryton, desde un tutorial publicado por la Cooperativa Devecoop y también con material proporcionado por un grupo de Bioingenieros que instalaron el servidor de Tryton en diferentes Hospitales de la provincia de Entre Rios para que puedan utilizar el software de gestión de salud GNU Health.

Y por último la capacitación a los usuarios, que en este caso sería los socios de la cooperativa.

## Guía ágil para instalación de Tryton con localización Argentina

### Instalación de Tryton 4.2 sobre Debian 8 Jessie

**Sistema Operativo:** Debian 8 Jessie

**Base de Datos:** Postgresql 9.4.19

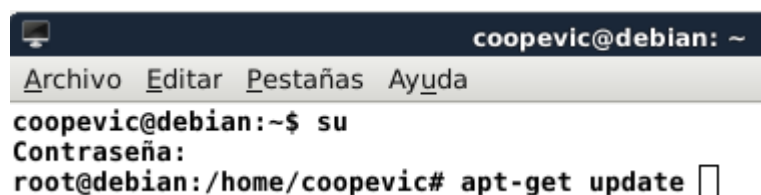
**ERP:** Tryton v4.2 Servidor



### Preparativos

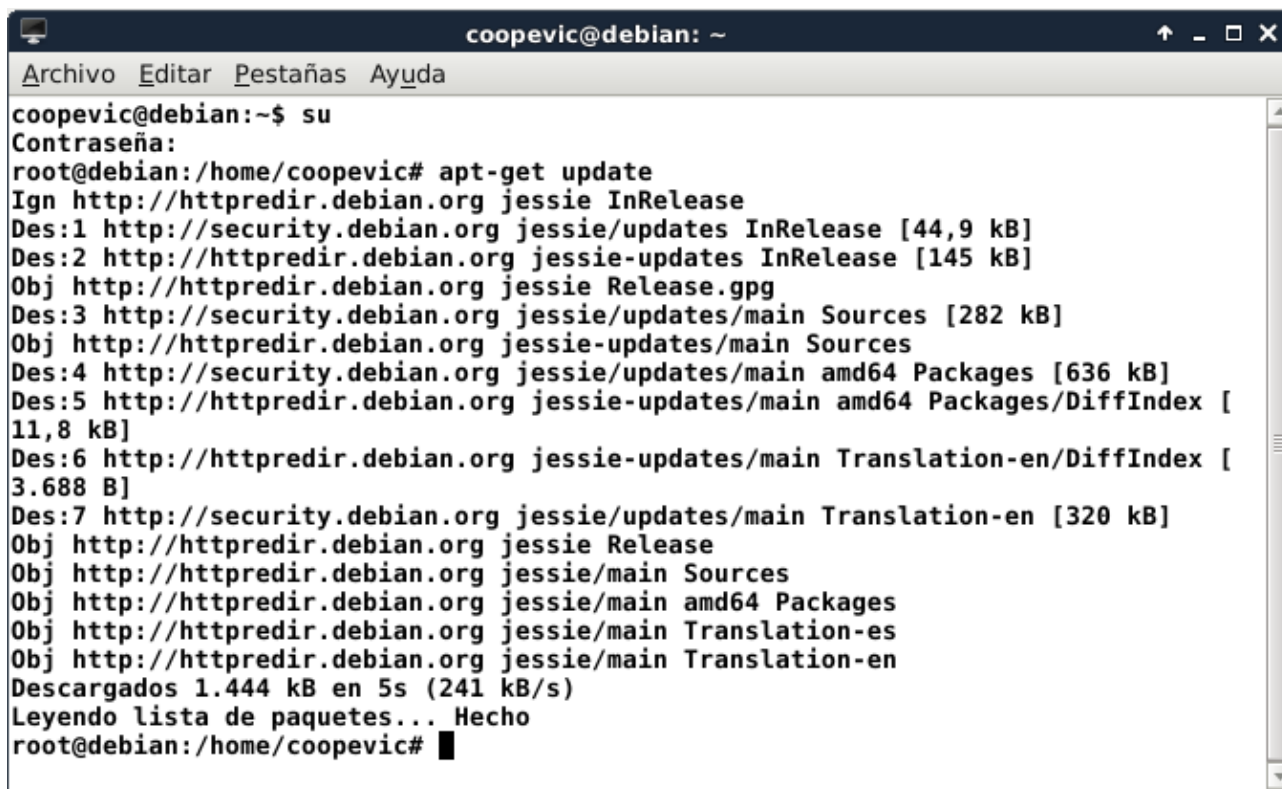
Primero actualizamos la lista de paquetes disponibles y sus versiones. Esta lista se saca de los servidores con repositorios que tenemos definidos en el ***sources.list***.

```
coopevic@debian:~$ su
Contraseña:
root@PC5:/home/coopevic# apt-get update
```



*Figura 1. Acceder como administrador y actualizamos paquetes*






```
coopevic@debian: ~  
Archivo Editar Pestañas Ayuda  
coopevic@debian:~$ su  
Contraseña:  
root@debian:/home/coopevic# apt-get update  
Ign http://httpredir.debian.org jessie InRelease  
Des:1 http://security.debian.org jessie/updates InRelease [44,9 kB]  
Des:2 http://httpredir.debian.org jessie-updates InRelease [145 kB]  
Obj http://httpredir.debian.org jessie Release.gpg  
Des:3 http://security.debian.org jessie/updates/main Sources [282 kB]  
Obj http://httpredir.debian.org jessie-updates/main Sources  
Des:4 http://security.debian.org jessie/updates/main amd64 Packages [636 kB]  
Des:5 http://httpredir.debian.org jessie-updates/main amd64 Packages/DiffIndex [11,8 kB]  
Des:6 http://httpredir.debian.org jessie-updates/main Translation-en/DiffIndex [3.688 B]  
Des:7 http://security.debian.org jessie/updates/main Translation-en [320 kB]  
Obj http://httpredir.debian.org jessie Release  
Obj http://httpredir.debian.org jessie/main Sources  
Obj http://httpredir.debian.org jessie/main amd64 Packages  
Obj http://httpredir.debian.org jessie/main Translation-es  
Obj http://httpredir.debian.org jessie/main Translation-en  
Descargados 1.444 kB en 5s (241 kB/s)  
Leyendo lista de paquetes... Hecho  
root@debian:/home/coopevic#
```

Figura 2. Actualizamos paquetes

Si queremos trabajar con **sudo**, en primer lugar configuraremos los **sudoers** en Debian. Para ello teclearemos la orden:

```
root@debian:/home/coopevic# visudo
```

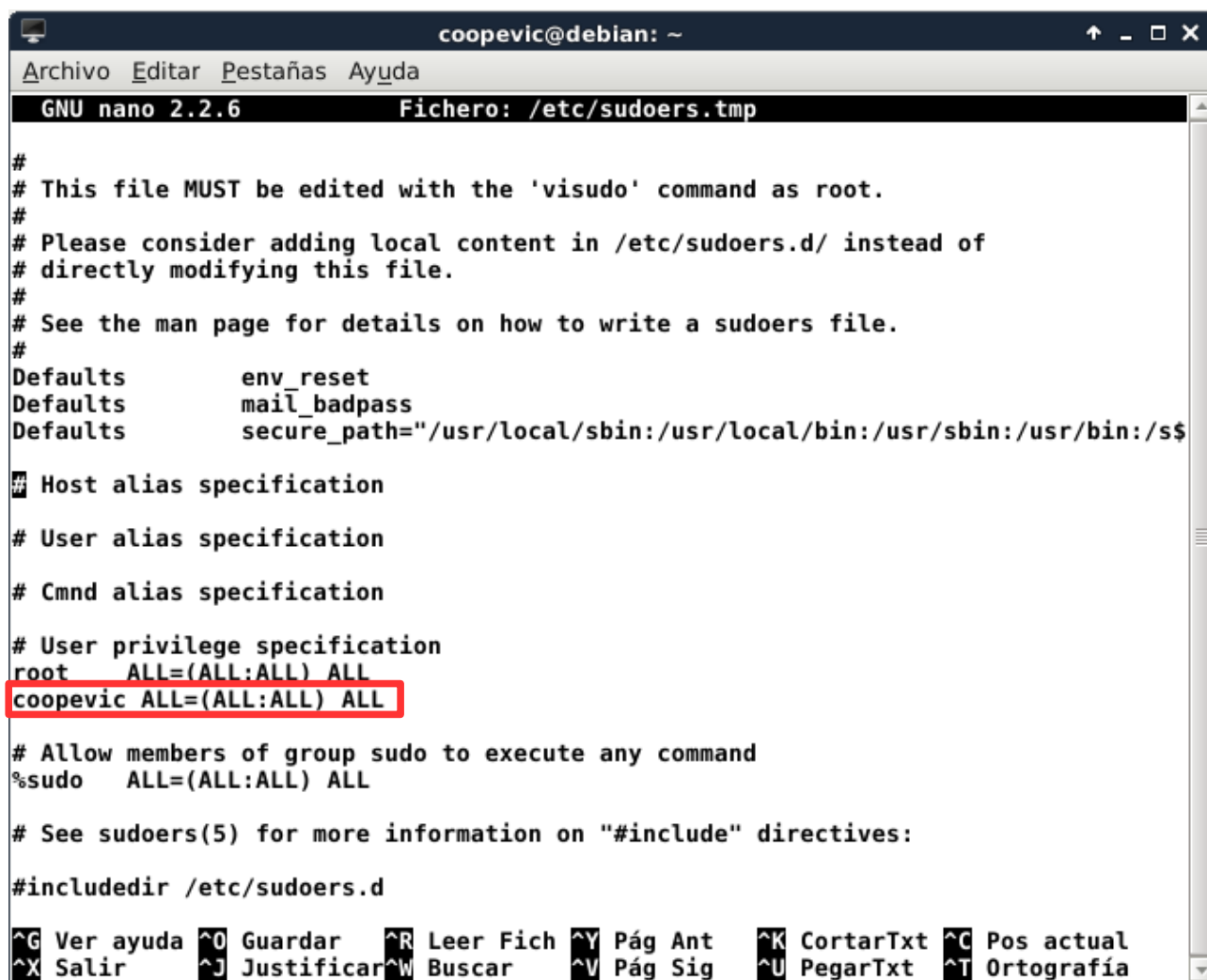


```
coopevic@debian: ~  
Archivo Editar Pestañas Ayuda  
root@debian:/home/coopevic# visudo
```

Figura 3. Configuramos sudoers para trabajar con sudo

se abrirá entonces el archivo **/etc/sudoers.tmp** y deberemos añadir nuestro usuario detrás del root:

```
coopevic ALL=(ALL:ALL) ALL
```



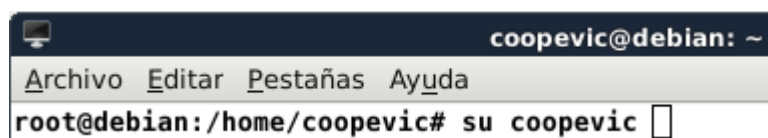
```
coopevic@debian: ~
Archivo Editar Pestañas Ayuda
GNU nano 2.2.6 Fichero: /etc/sudoers.tmp
#
# This file MUST be edited with the 'visudo' command as root.
#
# Please consider adding local content in /etc/sudoers.d/ instead of
# directly modifying this file.
#
# See the man page for details on how to write a sudoers file.
#
Defaults                env_reset
Defaults                mail_badpass
Defaults                secure_path="/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/s$
# Host alias specification
# User alias specification
# Cmnd alias specification
# User privilege specification
root    ALL=(ALL:ALL) ALL
coopevic ALL=(ALL:ALL) ALL
# Allow members of group sudo to execute any command
%sudo   ALL=(ALL:ALL) ALL
# See sudoers(5) for more information on "#include" directives:
#include /etc/sudoers.d
^G Ver ayuda ^O Guardar ^R Leer Fich ^Y Pág Ant ^K CortarTxt ^C Pos actual
^X Salir ^J Justificar ^W Buscar ^V Pág Sig ^U PegarTxt ^T Ortografía
```

Figura 4. Abrimos el archivo /etc/sudoers.tmp y añadimos nuestro usuario detrás del root:

De esta forma ya lo tenemos preparado para usar sudo.

Volvemos al usuario coopevic y trabajamos con sudo

**# su coopevic**



```
coopevic@debian: ~
Archivo Editar Pestañas Ayuda
root@debian:/home/coopevic# su coopevic
```

Figura 5. Volvemos al usuario coopevic y trabajamos con sudo

## Instalación de pip y virtualenvwrapper

### pip

Herramienta para instalar y gestionar paquetes de Python

### virtualenv

Herramienta para crear entornos controlados de Python. Permite tener diferentes versiones de algunas librerías y controlar las dependencias instaladas para cada entorno

### virtualenvwrapper

Conjunto de extensiones hechas sobre virtualenv para simplificar su uso.

```
$ sudo apt-get install python-pip
```



Figura 6. Instalamos python-pip, para instalar y gestionar paquetes de Python

```
$ pip install -user virtualenvwrapper
```

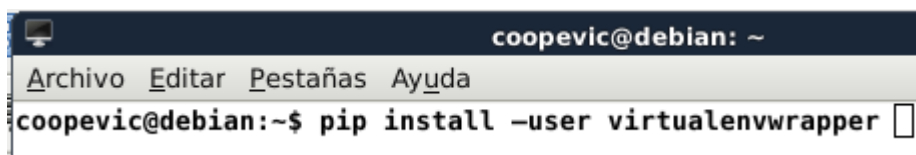


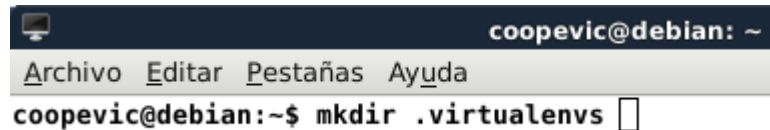
Figura 7. Instalamos virtualenvwrapper que es un Conjunto de extensiones hechas sobre virtualenv

## Configuración de virtualenvwrapper

Hay que indicarle al sistema cómo trabajar con virtualenvwrapper. Podemos crear una carpeta oculta llamada **.virtualenvs/**, en el home, y usarla para contener

los entornos creados.

### mkdir .virtualenvs

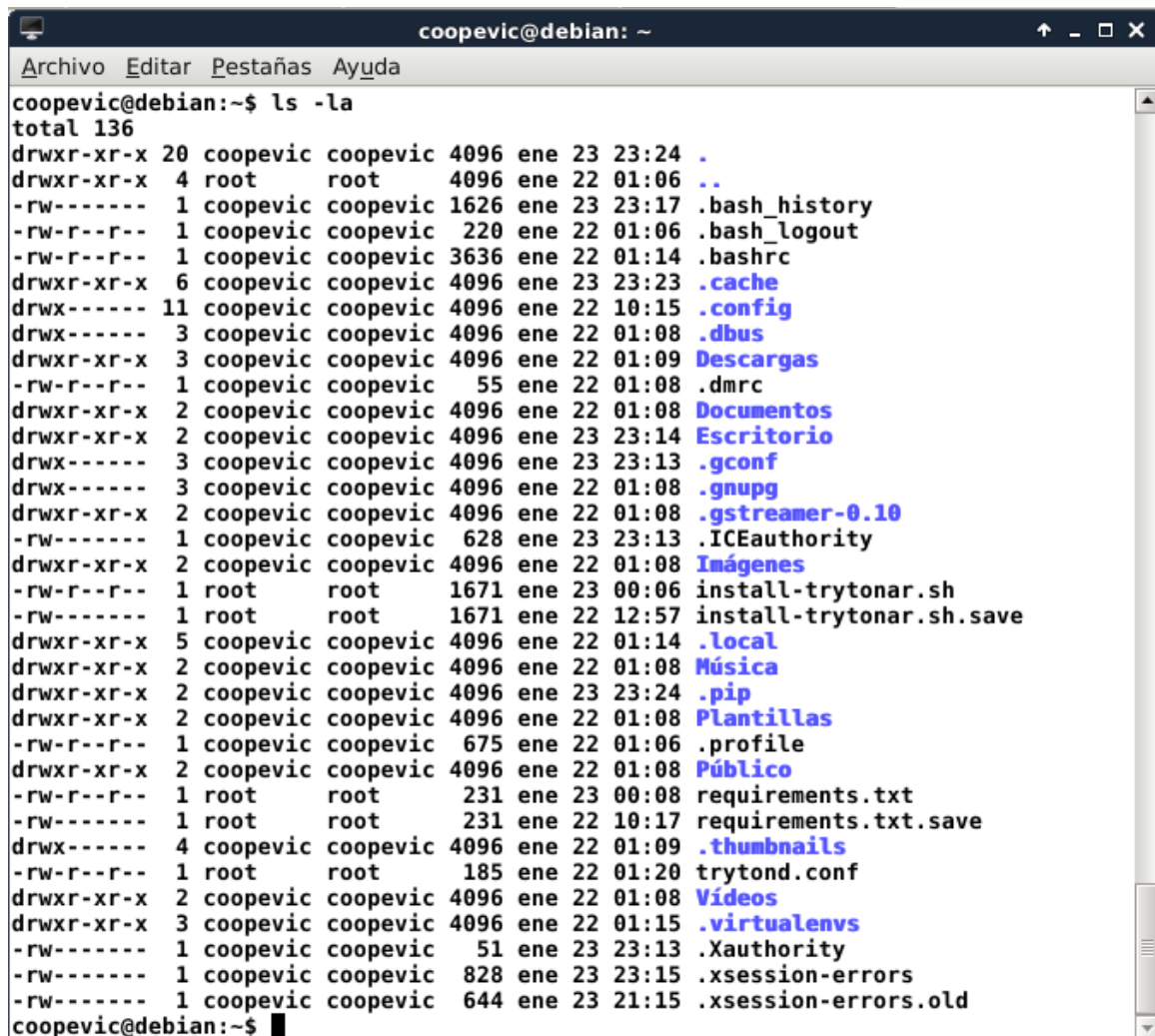


```
coopevic@debian: ~  
Archivo Editar Pestañas Ayuda  
coopevic@debian:~$ mkdir .virtualenvs
```

Figura 8. Creamos una carpeta oculta llamada virtualenvs dentro del home para contener los entornos creados

para ver la carpeta oculta use el el siguiente comando

### \$ ls -la



```
coopevic@debian: ~  
Archivo Editar Pestañas Ayuda  
coopevic@debian:~$ ls -la  
total 136  
drwxr-xr-x 20 coopevic coopevic 4096 ene 23 23:24 .  
drwxr-xr-x  4 root      root      4096 ene 22 01:06 ..  
-rw-r----- 1 coopevic coopevic 1626 ene 23 23:17 .bash_history  
-rw-r--r--  1 coopevic coopevic  220 ene 22 01:06 .bash_logout  
-rw-r--r--  1 coopevic coopevic 3636 ene 22 01:14 .bashrc  
drwxr-xr-x  6 coopevic coopevic 4096 ene 23 23:23 .cache  
drwx----- 11 coopevic coopevic 4096 ene 22 10:15 .config  
drwx-----  3 coopevic coopevic 4096 ene 22 01:08 .dbus  
drwxr-xr-x  3 coopevic coopevic 4096 ene 22 01:09 Descargas  
-rw-r--r--  1 coopevic coopevic   55 ene 22 01:08 .dmrc  
drwxr-xr-x  2 coopevic coopevic 4096 ene 22 01:08 Documentos  
drwxr-xr-x  2 coopevic coopevic 4096 ene 23 23:14 Escritorio  
drwx-----  3 coopevic coopevic 4096 ene 23 23:13 .gconf  
drwx-----  3 coopevic coopevic 4096 ene 22 01:08 .gnupg  
drwxr-xr-x  2 coopevic coopevic 4096 ene 22 01:08 .gststreamer-0.10  
-rw-r----- 1 coopevic coopevic  628 ene 23 23:13 .ICEauthority  
drwxr-xr-x  2 coopevic coopevic 4096 ene 22 01:08 Imágenes  
-rw-r--r--  1 root      root      1671 ene 23 00:06 install-trytonar.sh  
-rw-r----- 1 root      root      1671 ene 22 12:57 install-trytonar.sh.save  
drwxr-xr-x  5 coopevic coopevic 4096 ene 22 01:14 .local  
drwxr-xr-x  2 coopevic coopevic 4096 ene 22 01:08 Música  
drwxr-xr-x  2 coopevic coopevic 4096 ene 23 23:24 .pip  
drwxr-xr-x  2 coopevic coopevic 4096 ene 22 01:08 Plantillas  
-rw-r--r--  1 coopevic coopevic  675 ene 22 01:06 .profile  
drwxr-xr-x  2 coopevic coopevic 4096 ene 22 01:08 Público  
-rw-r--r--  1 root      root       231 ene 23 00:08 requirements.txt  
-rw-r----- 1 root      root       231 ene 22 10:17 requirements.txt.save  
drwx-----  4 coopevic coopevic 4096 ene 22 01:09 .thumbnails  
-rw-r--r--  1 root      root       185 ene 22 01:20 trytond.conf  
drwxr-xr-x  2 coopevic coopevic 4096 ene 22 01:08 Vídeos  
drwxr-xr-x  3 coopevic coopevic 4096 ene 22 01:15 .virtualenvs  
-rw-r----- 1 coopevic coopevic   51 ene 23 23:13 .Xauthority  
-rw-r----- 1 coopevic coopevic  828 ene 23 23:15 .xsession-errors  
-rw-r----- 1 coopevic coopevic  644 ene 23 21:15 .xsession-errors.old  
coopevic@debian:~$
```

Figura 9. para ver la carpeta oculta usamos el comando ls -la

Agregar en el archivo **/home/coopevic/.bashrc** las siguientes líneas:

```
$ sudo nano .bashrc
```

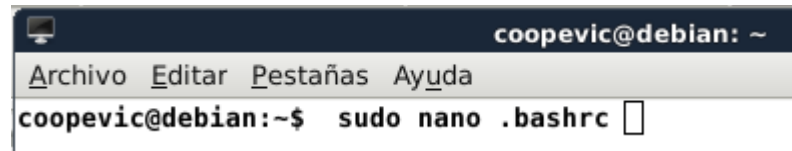


Figura 10. Ingresamos al archivo **/home/coopevic/.bashrc**

```
export WORKON_HOME=$HOME/.virtualenvs
export PATH=$PATH:$HOME/.local/bin
source $HOME/.local/bin/virtualenvwrapper.sh
```

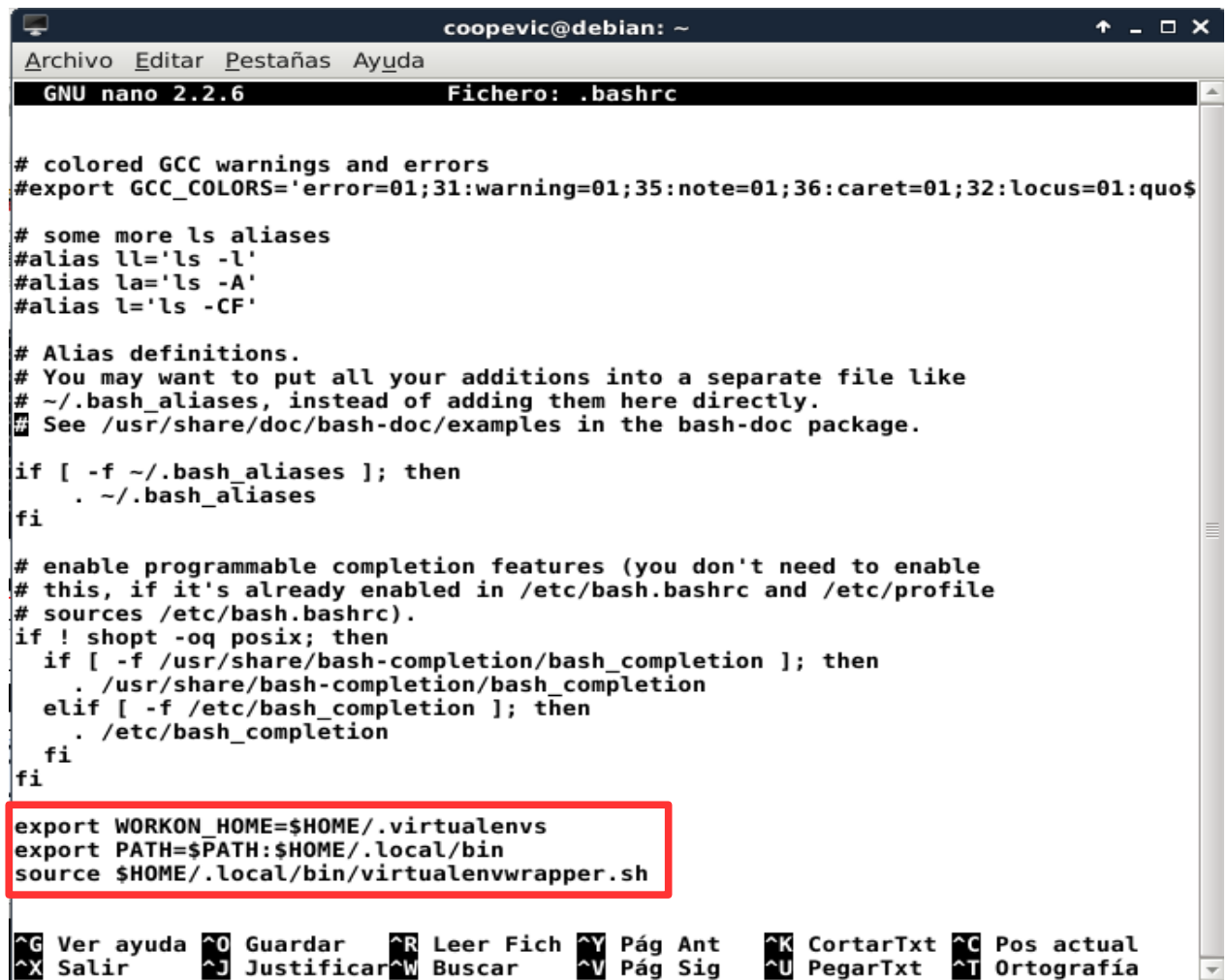
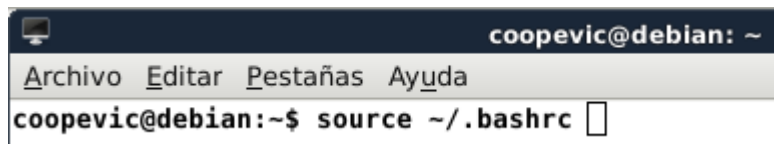


Figura 11. Modificamos el archivo **.bashrc**

hay que recargar el archivo **.bashrc (o reiniciar)**. Se puede hacer:

```
$ source ~/.bashrc
```



```
coopevic@debian: ~  
Archivo Editar Pestañas Ayuda  
coopevic@debian:~$ source ~/.bashrc
```

Figura 12. Recargamos el archivo `.bashrc`

De esta manera, en el home tendremos dos carpetas:

- **\$HOME/.local/:**

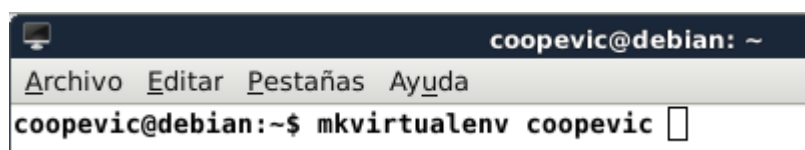
todo lo que se instale con pip con la opción `--user`

- **\$HOME/.virtualenvs/:**

los entornos virtuales creados con `virtualenv[wrapper]`

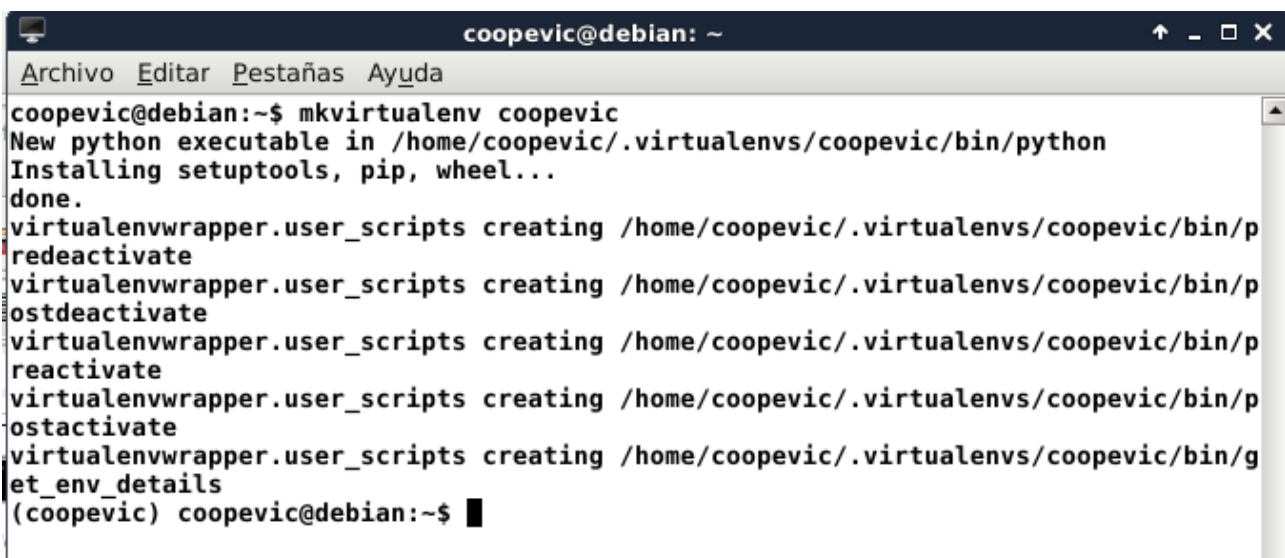
## Creación del entorno e instalación de Tryton

```
$ mkvirtualenv coopevic
```



```
coopevic@debian: ~  
Archivo Editar Pestañas Ayuda  
coopevic@debian:~$ mkvirtualenv coopevic
```

Figura 13. Creamos el entorno virtual `coopevic`



```
coopevic@debian: ~  
Archivo Editar Pestañas Ayuda  
coopevic@debian:~$ mkvirtualenv coopevic  
New python executable in /home/coopevic/.virtualenvs/coopevic/bin/python  
Installing setuptools, pip, wheel...  
done.  
virtualenvwrapper.user_scripts creating /home/coopevic/.virtualenvs/coopevic/bin/p  
redeactivate  
virtualenvwrapper.user_scripts creating /home/coopevic/.virtualenvs/coopevic/bin/p  
ostdeactivate  
virtualenvwrapper.user_scripts creating /home/coopevic/.virtualenvs/coopevic/bin/p  
reactivate  
virtualenvwrapper.user_scripts creating /home/coopevic/.virtualenvs/coopevic/bin/p  
ostactivate  
virtualenvwrapper.user_scripts creating /home/coopevic/.virtualenvs/coopevic/bin/g  
et_env_details  
(coopevic) coopevic@debian:~$
```

Figura 14. Creando el entorno virtual

## Descargar e instalar el servidor Tryton mediante el comando pip.

```
(coopevic)$ pip install 'trytond==4.2'
```



Figura 15. Descargamos e instalamos el servidor Tryton mediante el comando pip

## Instalación de módulos del repositorio oficial de Tryton

Se puede crear un archivo de requerimientos para instalar todos los módulos de tryton necesarios para la localización argentina en `requirements.txt`

```
$sudo nano requirements.txt
```

```
trytond_account==4.2
trytond_analytic_account==4.2
trytond_country==4.2
trytond_currency==4.2
trytond_party==4.2
trytond_company==4.2
trytond_bank==4.2
trytond_account_invoice==4.2
trytond_account_product==4.2
trytond_product==4.2
```

Usamos el comando pip para instalar los requerimientos

```
$ sudo pip install -r requirements.txt
```

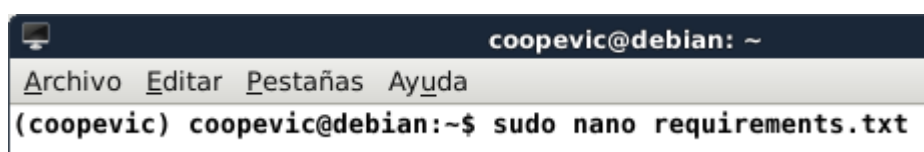
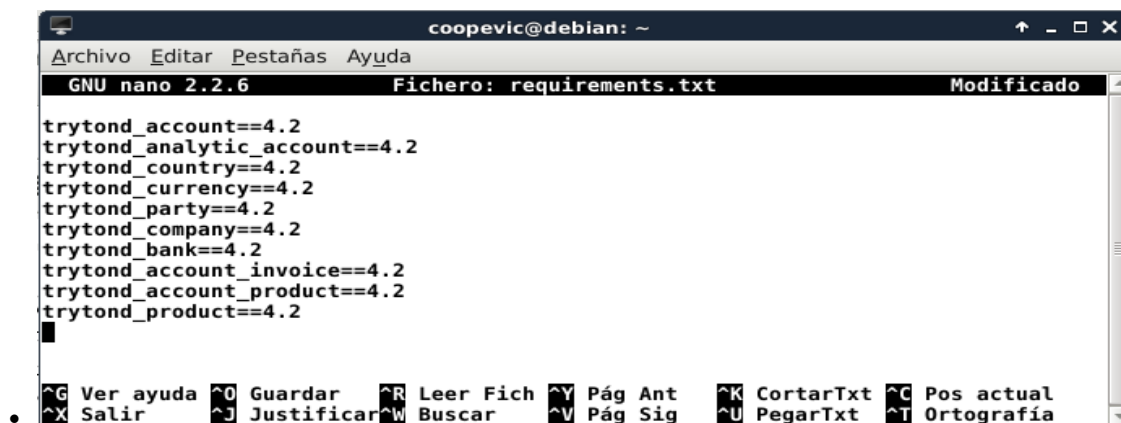


Figura 16. Creamos el archivo requirements.txt para instalar todos los módulos de tryton necesarios para la localización argentina



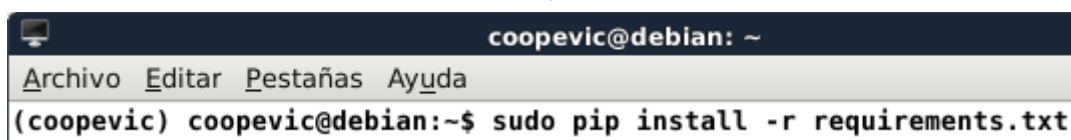


The screenshot shows a terminal window with the nano text editor open. The title bar reads 'coopevic@debian: ~'. The menu bar includes 'Archivo', 'Editar', 'Pestañas', and 'Ayuda'. The status bar at the top indicates 'GNU nano 2.2.6', 'Fichero: requirements.txt', and 'Modificado'. The editor contains the following text:

```
trytond_account==4.2
trytond_analytic_account==4.2
trytond_country==4.2
trytond_currency==4.2
trytond_party==4.2
trytond_company==4.2
trytond_bank==4.2
trytond_account_invoice==4.2
trytond_account_product==4.2
trytond_product==4.2
```

The bottom status bar shows various keyboard shortcuts: '^G Ver ayuda', '^X Salir', '^O Guardar', '^J Justificar', '^R Leer Fich', '^W Buscar', '^Y Pág Ant', '^V Pág Sig', '^K CortarTxt', '^U PegarTxt', '^C Pos actual', and '^T Ortografía'.

Figura 17. Archivo requirements.txt



The screenshot shows a terminal window with the title 'coopevic@debian: ~'. The menu bar includes 'Archivo', 'Editar', 'Pestañas', and 'Ayuda'. The command being executed is:

```
(coopevic) coopevic@debian:~$ sudo pip install -r requirements.txt
```

Figura 18. Instalamos los módulos del archivo requirements.txt

También se pueden instalar los módulos individualmente

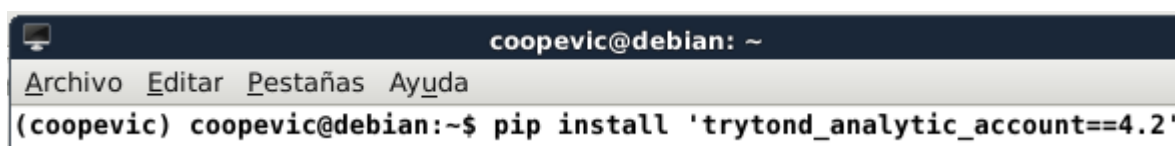
```
(coopevic)$ pip install 'trytond_account==4.2'
(coopevic)$ pip install 'trytond_analytic_account==4.2'
(coopevic)$ pip install 'trytond_country==4.2'
(coopevic)$ pip install 'trytond_currency==4.2'
(coopevic)$ pip install 'trytond_party==4.2'
(coopevic)$ pip install 'trytond_company==4.2'
(coopevic)$ pip install 'trytond_bank==4.2'
(coopevic)$ pip install 'trytond_account_invoice==4.2'
(coopevic)$ pip install 'trytond_account_product==4.2'
(coopevic)$ pip install 'trytond_product==4.2'
```



The screenshot shows a terminal window with the title 'coopevic@debian: ~'. The menu bar includes 'Archivo', 'Editar', 'Pestañas', and 'Ayuda'. The command being executed is:

```
(coopevic) coopevic@debian:~$ pip install 'trytond_account==4.2'
```

Figura 19. Instalamos el módulo trytond\_account



The screenshot shows a terminal window with the title 'coopevic@debian: ~'. The menu bar includes 'Archivo', 'Editar', 'Pestañas', and 'Ayuda'. The command being executed is:

```
(coopevic) coopevic@debian:~$ pip install 'trytond_analytic_account==4.2'
```

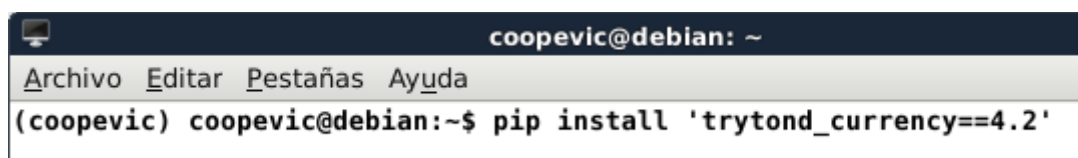
Figura 20. Instalamos el módulo trytond\_analytic\_account





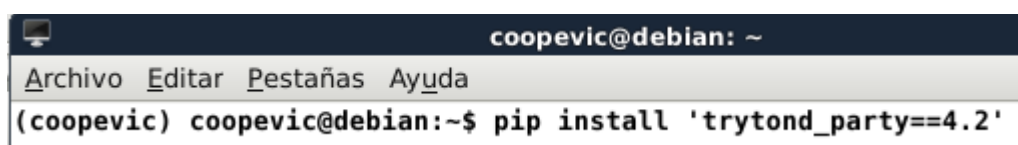
```
coopevic@debian: ~  
Archivo Editar Pestañas Ayuda  
(coopevic) coopevic@debian:~$ pip install 'trytond_country==4.2'
```

*Figura 21. Instalamos el módulo trytond\_country*



```
coopevic@debian: ~  
Archivo Editar Pestañas Ayuda  
(coopevic) coopevic@debian:~$ pip install 'trytond_currency==4.2'
```

*Figura 22. Instalamos el módulo trytond\_currency*



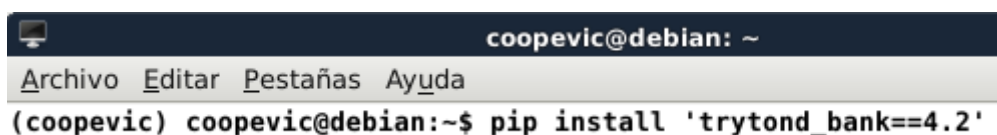
```
coopevic@debian: ~  
Archivo Editar Pestañas Ayuda  
(coopevic) coopevic@debian:~$ pip install 'trytond_party==4.2'
```

*Figura 23. Instalamos el módulo trytond\_party*



```
coopevic@debian: ~  
Archivo Editar Pestañas Ayuda  
(coopevic) coopevic@debian:~$ pip install 'trytond_company==4.2'
```

*Figura 24. Instalamos el módulo trytond\_company*



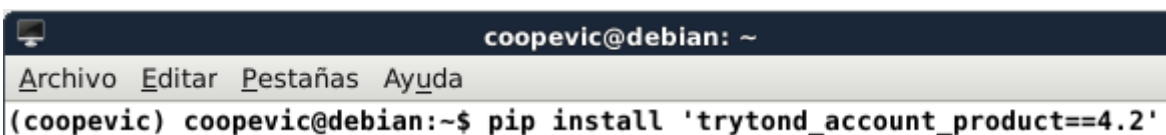
```
coopevic@debian: ~  
Archivo Editar Pestañas Ayuda  
(coopevic) coopevic@debian:~$ pip install 'trytond_bank==4.2'
```

*Figura 25. Instalamos el módulo trytond\_bank*



```
coopevic@debian: ~  
Archivo Editar Pestañas Ayuda  
(coopevic) coopevic@debian:~$ pip install 'trytond_account_invoice==4.2'
```

*Figura 26. Instalamos el módulo trytond\_account\_invoice*



```
coopevic@debian: ~  
Archivo Editar Pestañas Ayuda  
(coopevic) coopevic@debian:~$ pip install 'trytond_account_product==4.2'
```

*Figura 27. Instalamos el módulo trytond\_account\_product*



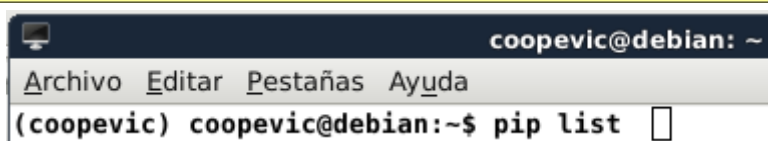
```

coopevic@debian: ~
Archivo Editar Pestañas Ayuda
(coopevic) coopevic@debian:~$ pip install 'trytond_product==4.2'
    
```

Figura 28. Instalamos el módulo trytond\_product

El comando `pip list` permite ver los paquetes instalados

(coopevic)\$ `pip list`



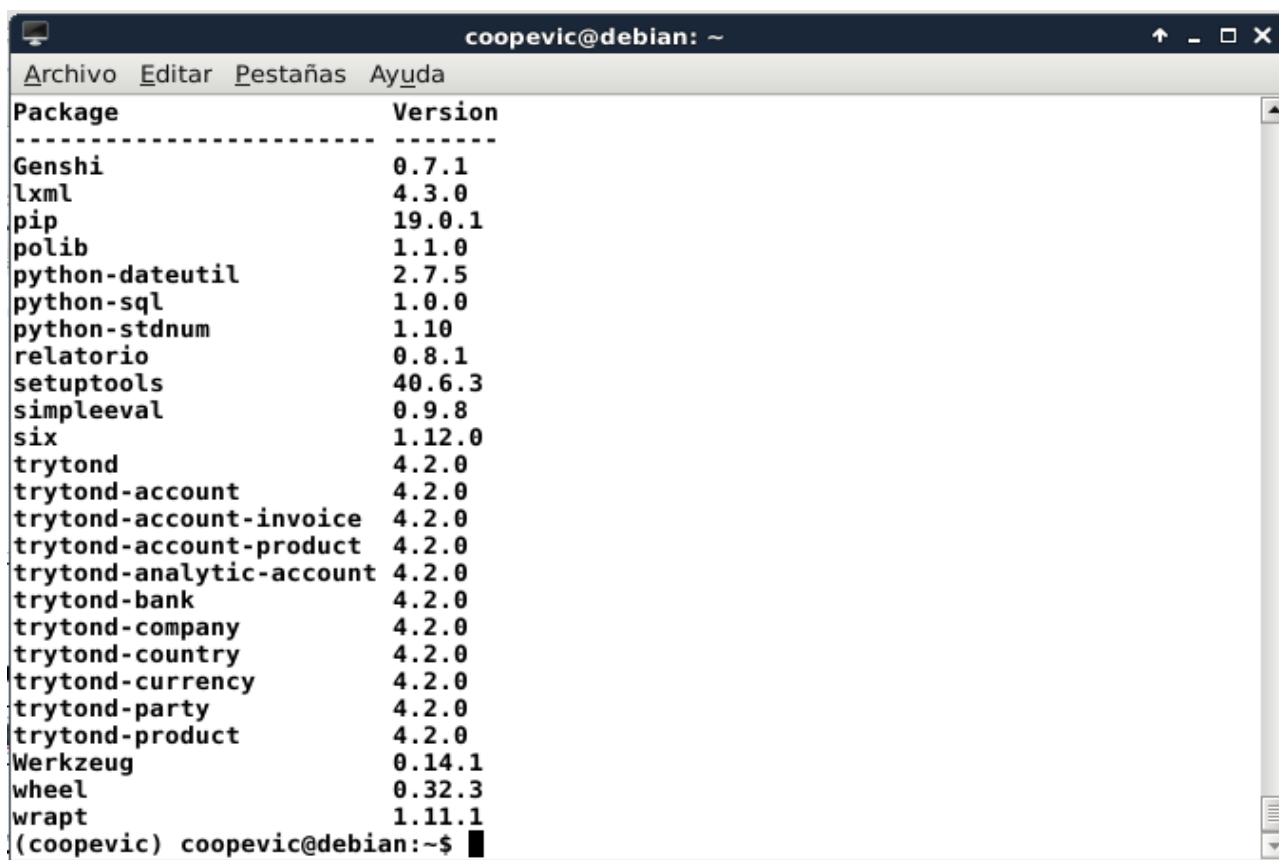
```

coopevic@debian: ~
Archivo Editar Pestañas Ayuda
(coopevic) coopevic@debian:~$ pip list
    
```

Figura 29. Con el comando `pip list` vemos los paquetes instalados

(coopevic) coopevic@debian:~\$ `pip list`

Package	Version
-----	-----
Genshi	0.7.1
lxml	4.3.0
pip	18.1
polib	1.1.0
python-dateutil	2.7.5
python-sql	1.0.0
python-stdnum	1.10
relatorio	0.8.1
setuptools	40.6.3
simpleeval	0.9.8
six	1.12.0
trytond	4.2.0
trytond-account	4.2.0
trytond-account-invoice	4.2.0
trytond-account-product	4.2.0
trytond-analytic-account	4.2.0
trytond-bank	4.2.0
trytond-company	4.2.0
trytond-country	4.2.0
trytond-currency	4.2.0
trytond-party	4.2.0
trytond-product	4.2.0
Werkzeug	0.14.1
wheel	0.32.3
wrap	1.11.0



The screenshot shows a terminal window titled 'coopevic@debian: ~'. The window contains a list of installed packages and their versions, organized into two columns: 'Package' and 'Version'. The packages listed are: Genshi (0.7.1), lxml (4.3.0), pip (19.0.1), polib (1.1.0), python-dateutil (2.7.5), python-sql (1.0.0), python-stdnum (1.10), relatorio (0.8.1), setuptools (40.6.3), simpleeval (0.9.8), six (1.12.0), trytond (4.2.0), trytond-account (4.2.0), trytond-account-invoice (4.2.0), trytond-account-product (4.2.0), trytond-analytic-account (4.2.0), trytond-bank (4.2.0), trytond-company (4.2.0), trytond-country (4.2.0), trytond-currency (4.2.0), trytond-party (4.2.0), trytond-product (4.2.0), Werkzeug (0.14.1), wheel (0.32.3), and wrapt (1.11.1). The terminal prompt is '(coopevic) coopevic@debian:~\$'.

Package	Version
Genshi	0.7.1
lxml	4.3.0
pip	19.0.1
polib	1.1.0
python-dateutil	2.7.5
python-sql	1.0.0
python-stdnum	1.10
relatorio	0.8.1
setuptools	40.6.3
simpleeval	0.9.8
six	1.12.0
trytond	4.2.0
trytond-account	4.2.0
trytond-account-invoice	4.2.0
trytond-account-product	4.2.0
trytond-analytic-account	4.2.0
trytond-bank	4.2.0
trytond-company	4.2.0
trytond-country	4.2.0
trytond-currency	4.2.0
trytond-party	4.2.0
trytond-product	4.2.0
Werkzeug	0.14.1
wheel	0.32.3
wrapt	1.11.1

Figura 30. Paquetes instalados con el comando pip

Para activar un entorno utilizamos el comando **workon**.

```
$ workon coopevic  
(coopevic)$ ....
```

Para remover un entorno utilizamos el comando **deactivate**.

```
(coopevic)$ deactivate
```

Para remover un entorno utilizamos el comando **rmvirtualenv**.

```
$rmvirtualenv coopevic
```

Otros comandos que podrían ser útiles

```
$lsvirtualenv // lista los entornos virtuales creados  
$cpvirtualenv mi_entorno mi_entorno2 // para copiar los entornos virtuales  
$mvvirtualenv // para mover o renombrar un entorno virtual
```

## El archivo de configuración trytond.conf

Los parámetros principales a configurar son los siguientes:

```
$ sudo nano trytond.conf
```

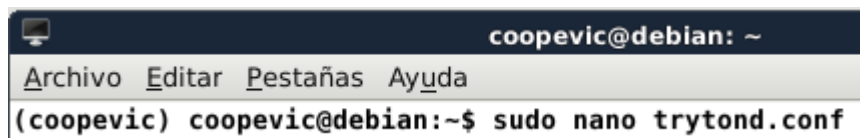


Figura 31. Accedemos al archivo de configuración trytond.conf

```
[jsonrpc]
listen=*:8000
data=/var/www/localhost/tryton
[database]
uri=postgresql://tryton:tryton@localhost:5432
path=/var/lib/tryton/data
[session]
timeout=3600
super_pwd=V6imlhDMI0fiY
```

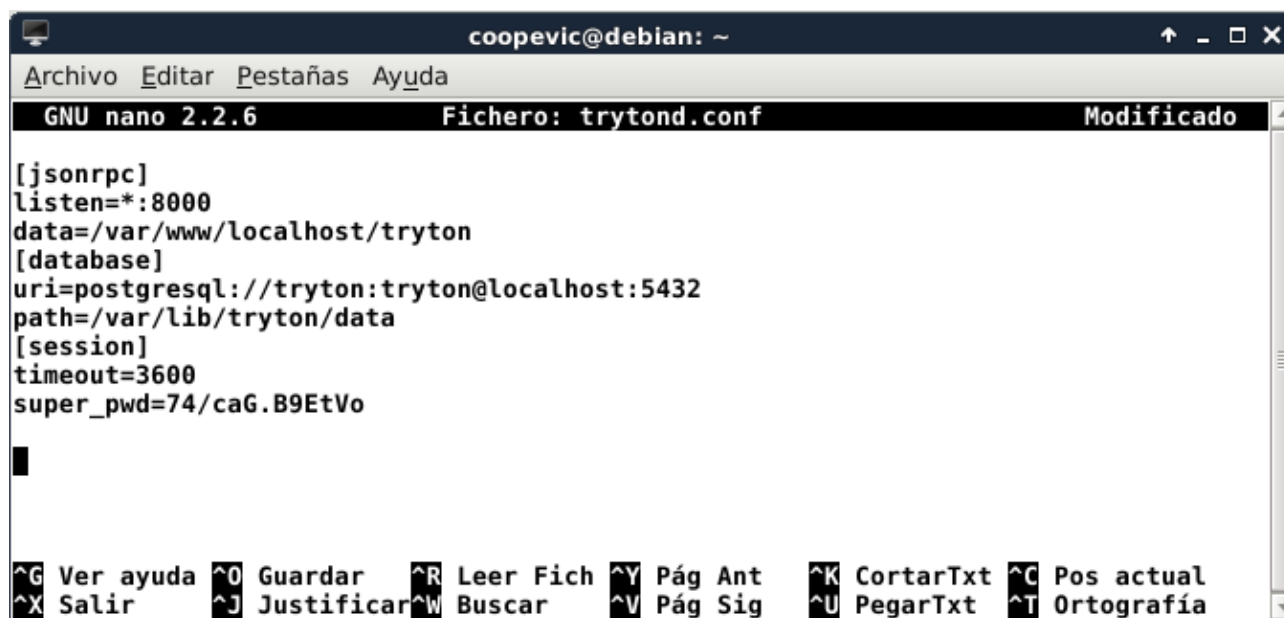


Figura 32. Modificamos el archivo de configuración tryton.conf

**jsonrpc:** define la interfaz de red.

**Listen:** define si se establecerá una conexión local o abierta (si se conectara con otros dispositivos)

**database:**

**path=/var/lib/tryton/data:** adonde van a parar los datos persistentes

El servidor PostgreSQL está escuchando en localhost en el puerto 5432

**session:** parametros de sesión

**timeout:** Tiempo de inactividad para volver a pedir la contraseña

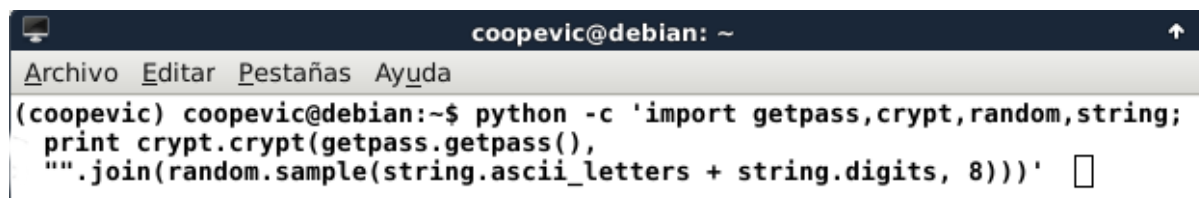
**super\_pwd:** Contraseña de super usuario encriptada.

## La contraseña de super usuario

En el archivo de configuración la contraseña de super usuario tryton debe estar encriptada.

Para encriptar la contraseña podemos ejecutar el siguiente comando en una terminal. El resultado que se obtiene es el que hay que agregar en el archivo de configuración, como valor de super\_pwd.

```
$ python -c 'import getpass,crypt,random,string; print crypt.crypt(getpass.getpass(),  
"".join(random.sample(string.ascii_letters + string.digits, 8)))'
```



*Figura 33. Ejecutamos este comando para encriptar la contraseña*

## Creación del usuario Postgresql cuya clave será “tryton”

En el archivo de configuración, necesitamos para tener acceso a la capa de la base de datos con un usuario de la misma con su correspondiente contraseña.

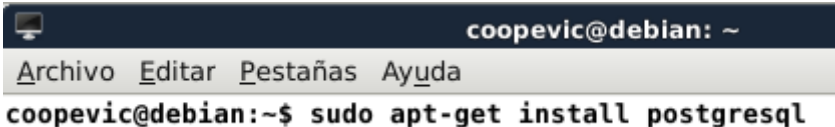
Recordando la línea de dicho archivo:

```
[...]  
[database]  
uri=postgresql://usuario:password@localhost:5432  
path=/var/lib/tryton/data  
[...]
```

Vemos que nuestro usuario de postgres se llamará usuario y su correspondiente password será password (valga la redundancia). Lo que sigue de @ corresponde a la localización de nuestra capa persistente, que en este caso es localhost, la misma máquina donde se este ejecutando el servidor trytond, que tiene el puerto 5432 escuchando, que es el que esta configurado por defecto en postgres.

Primero deberemos abrir un terminal e instalar postgresql

```
$sudo apt-get install postgresql
```

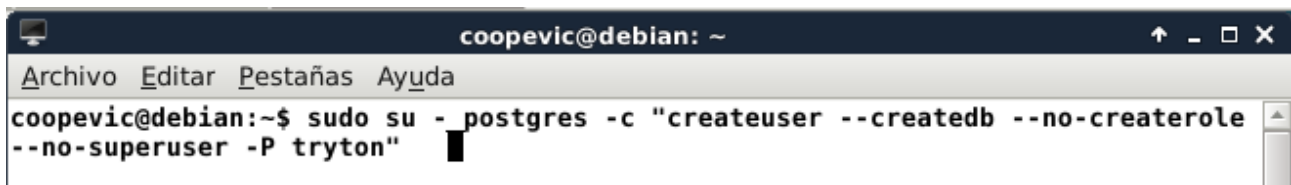


The screenshot shows a terminal window with a dark blue title bar containing a terminal icon and the text 'coopevic@debian: ~'. Below the title bar is a menu bar with 'Archivo', 'Editar', 'Pestañas', and 'Ayuda'. The main terminal area shows the prompt 'coopevic@debian:~\$' followed by the command 'sudo apt-get install postgresql'.

*Figura 34. Instalamos postgresql*

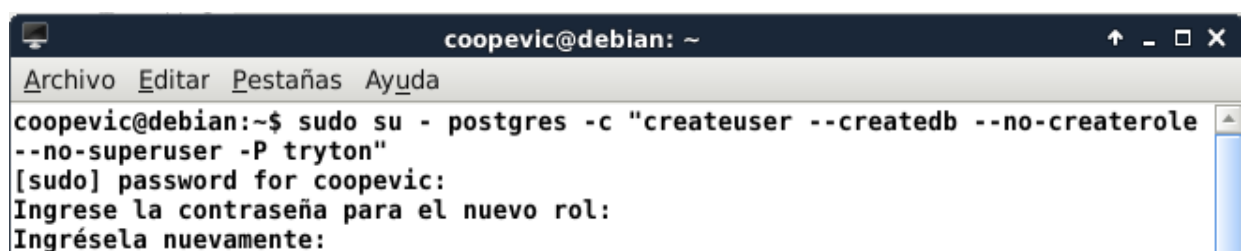
*Y a continuación, tendremos que ejecutar el siguiente comando, que creara el usuario, donde se nos pedirá luego definir su contraseña.*

```
$sudo su - postgres -c "createuser --createdb --no-createrole --no-superuser -P tryton"
```



```
coopevic@debian: ~  
Archivo Editar Pestañas Ayuda  
coopevic@debian:~$ sudo su - postgres -c "createuser --createdb --no-createrole --no-superuser -P tryton"
```

Figura 35. Creamos el usuario de Postgres



```
coopevic@debian: ~  
Archivo Editar Pestañas Ayuda  
coopevic@debian:~$ sudo su - postgres -c "createuser --createdb --no-createrole --no-superuser -P tryton"  
[sudo] password for coopevic:  
Ingrese la contraseña para el nuevo rol:  
Ingrésela nuevamente:
```

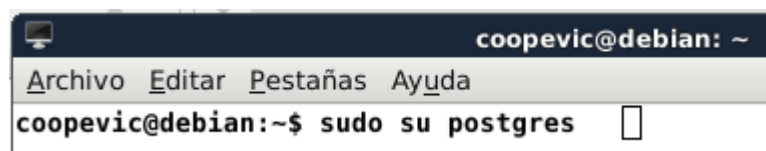
Figura 36. Nos solicita contraseña para el nuevo rol

Antes de la versión de Tryton 4.2, se podía crear la base de datos desde tryton. Desde la versión 4.2, se debe crear con anterioridad con Postgres, de la siguiente manera:

### Crear base de datos

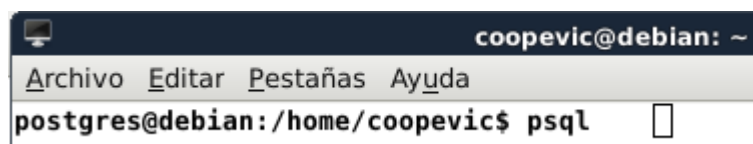
Ingresamos a postgres.

```
$ sudo su postgres  
$ psql
```



```
coopevic@debian: ~  
Archivo Editar Pestañas Ayuda  
coopevic@debian:~$ sudo su postgres
```

Figura 37. Cambiamos al usuario postgres

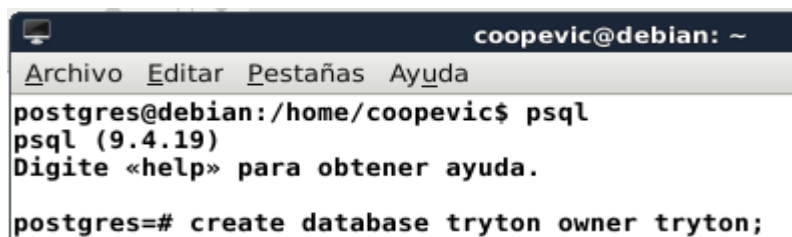


```
coopevic@debian: ~  
Archivo Editar Pestañas Ayuda  
postgres@debian:/home/coopevic$ psql
```

Figura 38. Ingresamos a postgres

Creamos la base de datos.

```
postgres=# create database tryton owner tryton;  
CREATE DATABASE
```

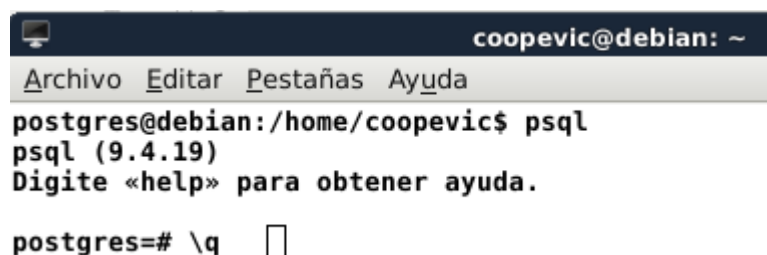
A terminal window titled 'coopevic@debian: ~' with a menu bar containing 'Archivo', 'Editar', 'Pestañas', and 'Ayuda'. The terminal shows the command 'psql' being executed, which opens the 'psql (9.4.19)' prompt. The user then enters 'create database tryton owner tryton;' and the prompt returns to 'postgres=#'.

```
coopevic@debian: ~  
Archivo  Editar  Pestañas  Ayuda  
postgres@debian:/home/coopevic$ psql  
psql (9.4.19)  
Digite «help» para obtener ayuda.  
postgres=# create database tryton owner tryton;  
postgres=#
```

*Figura 39. Creamos la base de datos*

Salimos de Postgres.

```
\q #Salimos de psql
```

A terminal window titled 'coopevic@debian: ~' with a menu bar containing 'Archivo', 'Editar', 'Pestañas', and 'Ayuda'. The terminal shows the command 'psql' being executed, which opens the 'psql (9.4.19)' prompt. The user then enters '\q' and the prompt returns to 'postgres=#'.

```
coopevic@debian: ~  
Archivo  Editar  Pestañas  Ayuda  
postgres@debian:/home/coopevic$ psql  
psql (9.4.19)  
Digite «help» para obtener ayuda.  
postgres=# \q
```

*Figura 40. Salimos de Postgres*

Salimos del usuario postgres.

```
$ exit
```

A terminal window titled 'coopevic@debian: ~' with a menu bar containing 'Archivo', 'Editar', 'Pestañas', and 'Ayuda'. The terminal shows the command 'exit' being executed, which returns the user to the shell prompt '\$'.

```
coopevic@debian: ~  
Archivo  Editar  Pestañas  Ayuda  
postgres@debian:/home/coopevic$ exit
```

*Figura 41. Salimos del usuario Postgres*



## Instalación las dependencias:

**Psycopg2** es un adaptador PostgreSQL para el lenguaje Python implementado utilizando libpq, la librería oficial del cliente PostgreSQL

```
sudo apt-get install python-dateutil python-psycopg2
```

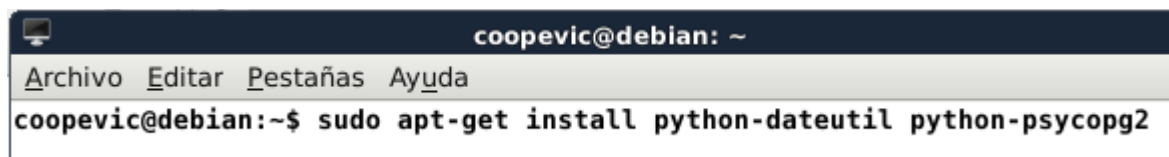


Figura 42. Instalamos Psycopg2 que es un adaptador de PostgreSQL para el lenguaje Python

## Instalación de git para poder descargar los módulos de github

Git es un software de control de versiones, necesario para poder descargar los módulos desde github.

```
sudo apt-get install git
```



Figura 43. Instalamos Git que es un software de control de versiones

## Instalación de las dependencias en el entorno virtual

```
$(coopevic) pip install psycopg2-binary
```



Figura 44. Instalamos dependencias en el entorno virtual

## Descarga e instalación de requerimientos de pyafipws

SWIG: es una herramienta de desarrollo de software que conecta programas escritos en C y C ++ con una variedad de lenguajes de programación de alto nivel. SWIG se utiliza con diferentes tipos de idiomas de destino, incluidos los lenguajes de script comunes como Javascript, Perl, PHP, Python, Tcl y Ruby.

```
sudo apt-get install swig
```



```
coopevic@debian: ~  
Archivo Editar Pestañas Ayuda  
(coopevic) coopevic@debian:~$ sudo apt-get install swig
```

*Figura 45. Instalación de swig*

## Instalación en el directotio raiz de httpplib2 y M2Crypto

**httpplib2:** El módulo httpplib2 es una completa librería de clientes HTTP que maneja el almacenamiento en caché, keep-alive, compresión, redirecciones y muchos tipos de autenticación.

**M2Crypto:** un kit de herramientas SSL y criptográfico de Python

```
$ sudo apt-get install python-httpplib2
```



```
coopevic@debian: /  
Archivo Editar Pestañas Ayuda  
(coopevic) coopevic@debian:/$ sudo apt-get install python-httpplib2
```

*Figura 46. Instalación de python-httpplib2*

```
$ sudo apt-get install python-m2crypto
```



```
coopevic@debian: /  
Archivo Editar Pestañas Ayuda  
(coopevic) coopevic@debian:/$ sudo apt-get install python-m2crypto
```

Figura 47. Instalación de python-m2crypto

### Copiar M2Crypto\* a site\_package

```
cp -r /usr/lib/python2.7/dist-packages/M2Crypto*  
~/.virtualenvs/coopevic/lib/python2.7/site-packages/
```



```
coopevic@debian: /  
Archivo Editar Pestañas Ayuda  
(coopevic) coopevic@debian:/$ cp -r /usr/lib/python2.7/dist-packages/M2Crypto* ~  
/.virtualenvs/tryton/lib/python2.7/site-packages/
```

Figura 48. Copiamos M2Crypto\* a site\_package

### Cambiar al directorio site-packages del virtualenv

```
cd ~/.virtualenvs/coopevic/lib/python2.7/site-packages/
```

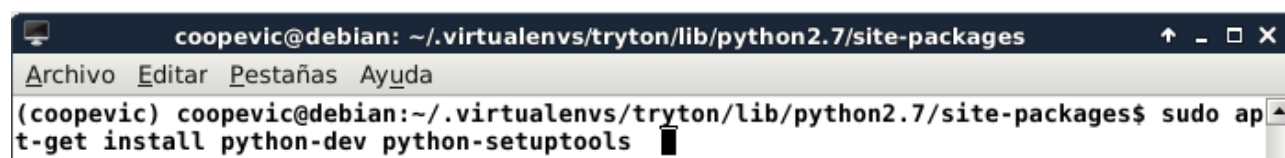


```
coopevic@debian: /  
Archivo Editar Pestañas Ayuda  
(coopevic) coopevic@debian:/$ cd ~/.virtualenvs/tryton/lib/python2.7/site-packages/
```

Figura 49 Cambiamos al directorio site-packages

### Antes de instalar Pillow, asegurarse de haber instalado las dependencias:

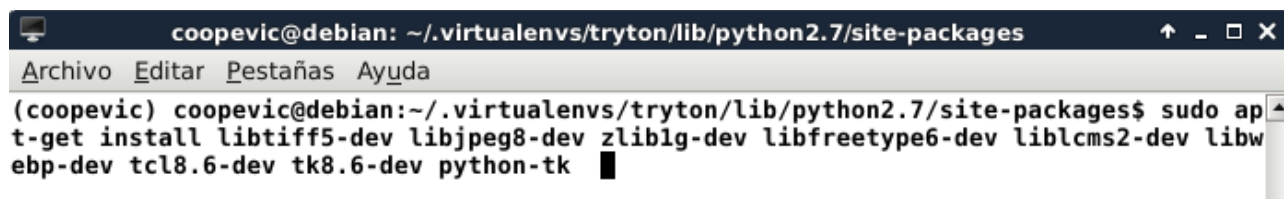
```
$ sudo apt-get install python-dev python-setuptools
```



```
coopevic@debian: ~/.virtualenvs/tryton/lib/python2.7/site-packages  
Archivo Editar Pestañas Ayuda  
(coopevic) coopevic@debian:~/.virtualenvs/tryton/lib/python2.7/site-packages$ sudo ap  
t-get install python-dev python-setuptools
```

Figura 50. Instalamos dependencias.

```
$ sudo apt-get install libtiff5-dev libjpeg8-dev zlib1g-dev libfreetype6-dev liblcms2-  
dev libwebp-dev tcl8.6-dev tk8.6-dev python-tk
```



```
coopevic@debian: ~/.virtualenvs/tryton/lib/python2.7/site-packages
Archivo Editar Pestañas Ayuda
(coopevic) coopevic@debian:~/.virtualenvs/tryton/lib/python2.7/site-packages$ sudo apt-get install libtiff5-dev libjpeg8-dev zlib1g-dev libfreetype6-dev liblcms2-dev libwebp-dev tcl8.6-dev tk8.6-dev python-tk
```

Figura 51. Instalamos dependencias

## Instalación de Pillow

**Pillow:** Biblioteca de imágenes de Python

```
pip install "Pillow>=2.8.1"
```



```
coopevic@debian: ~/.virtualenvs/tryton/lib/python2.7/site-packages
Archivo Editar Pestañas Ayuda
(coopevic) coopevic@debian:~/.virtualenvs/tryton/lib/python2.7/site-packages$ pip install "Pillow>=2.8.1"
```

Figura 52. Instalamos biblioteca de imágenes pillow

## Descarga e instalación de las dependencias PySimpleSoap y suds:

**PySimpleSoap:** es una librería que se desarrolló inicialmente para comunicarse con los servicios web SOAP de Factura Electrónica de AFIP

```
pip install "PySimpleSOAP==1.08.8"
```



```
coopevic@debian: ~/.virtualenvs/tryton/lib/python2.7/site-packages
Archivo Editar Pestañas Ayuda
(coopevic) coopevic@debian:~/.virtualenvs/tryton/lib/python2.7/site-packages$ pip install "PySimpleSOAP==1.08.8"
```

Figura 53. Instalamos PySimpleSOAP

```
pip install "suds>=0.4"
```



```
coopevic@debian: ~/.virtualenvs/tryton/lib/python2.7/site-packages
Archivo Editar Pestañas Ayuda
(coopevic) coopevic@debian:~/.virtualenvs/tryton/lib/python2.7/site-packages$ pip install "suds>=0.4"
```

Figura 54. Instalamos suds

## Instalar módulo pyafipws

El módulo *pyafipws* es empleado por la localización argentina para trabajar principalmente con la facturación electrónica exigida por AFIP en Argentina. También se utiliza para generar el código de barras del CAE.

Para instalarlo, primero clonamos el módulo desde github:

```
git clone https://github.com/reingart/pyafipws.git
```



Figura 55. Instalamos el módulo pyafipws

```
cd pyafipws
```

```
python setup.py install
```

## Otra forma de descargar e instalar pysimplesoap

```
git clone https://github.com/pysimplesoap/pysimplesoap.git
```

```
cd pysimplesoap
```

```
python setup.py install
```

## Cambiando al directorio de módulos de tryton

```
cd ~/.virtualenvs/coopevic/lib/python2.7/site-packages/trytond/modules
```

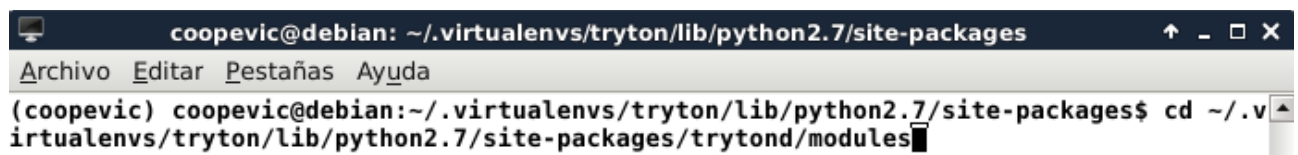


Figura 56. Cambiando al módulo de directorio de tryton

## Instalación de los módulos de trytonar y de cooperativas de Gcoop

Clonación de los módulo de Tryton Argentina y de Gcoop

```
git clone https://github.com/tryton-ar/account_ar.git
git clone https://github.com/tryton-ar/account_check_ar.git
git clone https://github.com/tryton-ar/account_invoice_ar.git
git clone https://github.com/tryton-ar/account_retencion_ar.git
git clone https://github.com/tryton-ar/account_voucher_ar.git
git clone https://github.com/tryton-ar/bank_ar.git
git clone https://github.com/tryton-ar/party_ar.git
git clone https://github.com/gcoop-libre/account_coop_ar.git
git clone https://github.com/gcoop-libre/trytond-cooperative_ar.git
git clone https://github.com/gcoop-libre/trytond-analytic_account_chart_template.git
```

## Renombrar las siguientes carpetas de módulos

```
trytond-cooperative_ar
trytond-analytic_account_chart_template
```

```
mv trytond-cooperative_ar cooperative_ar
mv trytond-analytic_account_chart_template analytic_account_chart_template
```

## Cambiar a la rama 4.2 e instalar los módulos de python

```
cd account_ar && git checkout '4.2'
sudo python setup.py install
cd account_check_ar && git checkout '4.2'
sudo python setup.py install
cd account_coop_ar && git checkout '4.2'
sudo python setup.py install
cd account_invoice_ar && git checkout '4.2'
sudo python setup.py install
cd account_retencion_ar && git checkout '4.2'
sudo python setup.py install
cd account_voucher_ar && git checkout '4.2'
sudo python setup.py install
cd bank_ar && git checkout '4.2'
sudo python setup.py install
cd cooperative_ar && git checkout '4.2'
sudo python setup.py install
cd party_ar && git checkout '4.2'
sudo python setup.py install
```

## Otra forma de Instalar los módulos, es usando un script

Creamos el script `install-trytonar.sh`

```
$ sudo nano install-trytonar.sh
```

```
#!/usr/bin/env bash
TRYTON_VERSION='4.2'
echo "Cambiando al directorio de módulos de tryton"
cd ~/.virtualenvs/tryton/lib/python2.7/site-packages/trytond/modules
echo "Clonando módulo de Tryton Argentina"
git clone https://github.com/tryton-ar/account_ar.git
git clone https://github.com/tryton-ar/account_check_ar.git
git clone https://github.com/tryton-ar/account_invoice_ar.git
git clone https://github.com/tryton-ar/account_retencion_ar.git
git clone https://github.com/tryton-ar/account_voucher_ar.git
git clone https://github.com/tryton-ar/bank_ar.git
git clone https://github.com/tryton-ar/party_ar.git
git clone https://github.com/gcoop-libre/account_coop_ar.git
git clone https://github.com/gcoop-libre/trytond-cooperative_ar.git
git clone https://github.com/gcoop-libre/trytond-analytic_account_chart_template.git
echo "Cambiando a la rama $TRYTON_VERSION"
cd account_ar && git checkout $TRYTON_VERSION
cd ../account_check_ar && git checkout $TRYTON_VERSION
cd ../account_coop_ar && git checkout $TRYTON_VERSION
cd ../account_invoice_ar && git checkout $TRYTON_VERSION
cd ../account_retencion_ar && git checkout $TRYTON_VERSION
cd ../account_voucher_ar && git checkout $TRYTON_VERSION
cd ../bank_ar && git checkout $TRYTON_VERSION
cd ../cooperative_ar && git checkout $TRYTON_VERSION
cd ../party_ar && git checkout $TRYTON_VERSION
echo "Instalando pyafipws....."
echo "Instalando requerimientos de pyafipws"
pip install "Pillow>=2.8.1" "M2Crypto>=0.22.3" "PySimpleSOAP==1.08.8"
"httpplib2>=0.9.1" "suds>=0.4"
echo "Cambiando al directorio site-packages del virtualenv"
cd ~/.virtualenvs/tryton/lib/python2.7/site-packages/
echo "Clonando pyafipws"
git clone https://github.com/reingart/pyafipws.git
```



## Script install-trytonar.sh



```
coopevic@debian: ~/.virtualenvs/tryton/lib/python2.7/site-packages/trytond/modules
GNU nano 2.2.6 Fichero: install-trytonar.sh Modificado
#!/usr/bin/env bash
TRYTON_VERSION='4.2'
echo "Cambiando al directorio de módulos de tryton"
cd ~/.virtualenvs/tryton/lib/python2.7/site-packages/trytond/modules
echo "Clonando módulo de Tryton Argentina"
git clone https://github.com/tryton-ar/account_ar.git
git clone https://github.com/tryton-ar/account_check_ar.git
git clone https://github.com/tryton-ar/account_invoice_ar.git
git clone https://github.com/tryton-ar/account_retencion_ar.git
git clone https://github.com/tryton-ar/account_voucher_ar.git
git clone https://github.com/tryton-ar/bank_ar.git
git clone https://github.com/tryton-ar/party_ar.git
git clone https://github.com/gcoop-libre/account_coop_ar.git
git clone https://github.com/gcoop-libre/trytond-cooperative_ar.git
git clone https://github.com/gcoop-libre/trytond-analytic_account_chart_template.git
echo "Cambiando a la rama $TRYTON_VERSION"
cd account_ar && git checkout $TRYTON_VERSION
cd ../account_check_ar && git checkout $TRYTON_VERSION
cd ../account_coop_ar && git checkout $TRYTON_VERSION
cd ../account_invoice_ar && git checkout $TRYTON_VERSION
cd ../account_retencion_ar && git checkout $TRYTON_VERSION
cd ../account_voucher_ar && git checkout $TRYTON_VERSION
cd ../bank_ar && git checkout $TRYTON_VERSION
cd ../cooperative_ar && git checkout $TRYTON_VERSION
cd ../party_ar && git checkout $TRYTON_VERSION
echo "Instalando pyafipws....."
echo "Instalando requerimientos de pyafipws"
pip install "Pillow>=2.8.1" "M2Crypto>=0.22.3" "PySimpleSOAP==1.08.8" "httplib2>=0.9.1" "suds>=0.4"
echo "Cambiando el directorio site-packages del virtualenv"
cd ~/.virtualenvs/tryton/lib/python2.7/site-packages/
echo "Clonando pyafipws"
git clone https://github.com/reingart/pyafipws.git

^G Ver ayuda ^O Guardar ^R Leer Fich ^Y Pág Ant ^K CortarTxt ^C Pos actual
^X Salir ^J Justificar ^W Buscar ^V Pág Sig ^U PegarTxt ^T Ortografía
```

Figura 57. Script install-trytonar.sh



## Inicializar la base de datos

```
(coopevic) trytond-admin -d tryton -c /home/coopevic/trytond.conf  
--all --verbose
```

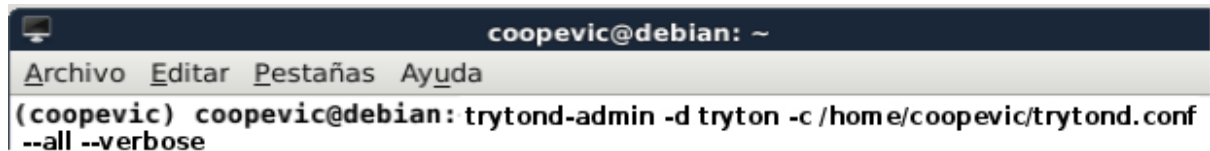


Figura 58. Inicializando la base de datos

## Arrancamos el servidor!

```
$ workon coopevic  
(coopevic)$ trytond -c /home/coopevic/trytond.conf -v  
[2017-05-12 12:16:16,718] INFO trytond.server using  
RUTA_AL_TRYTOND_CONF/trytond.conf as configuration file  
[2017-05-12 12:16:16,718] INFO trytond.server initialising distributed objects  
services  
[2017-05-12 12:16:16,745] INFO trytond.server starting JSON-RPC protocol on  
*:8100
```

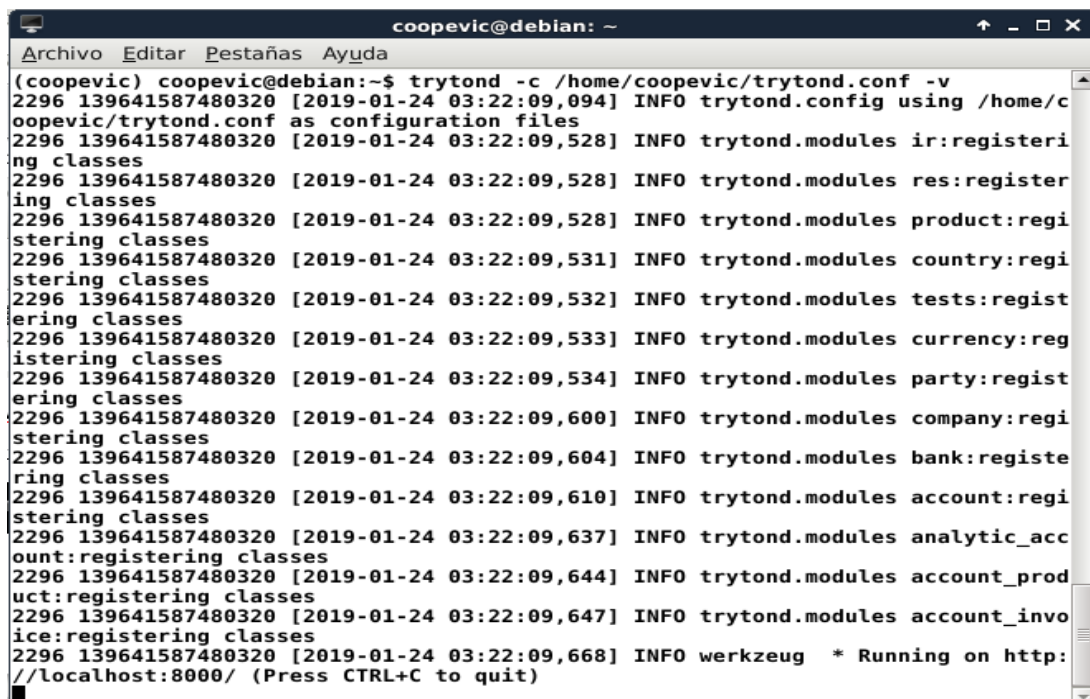
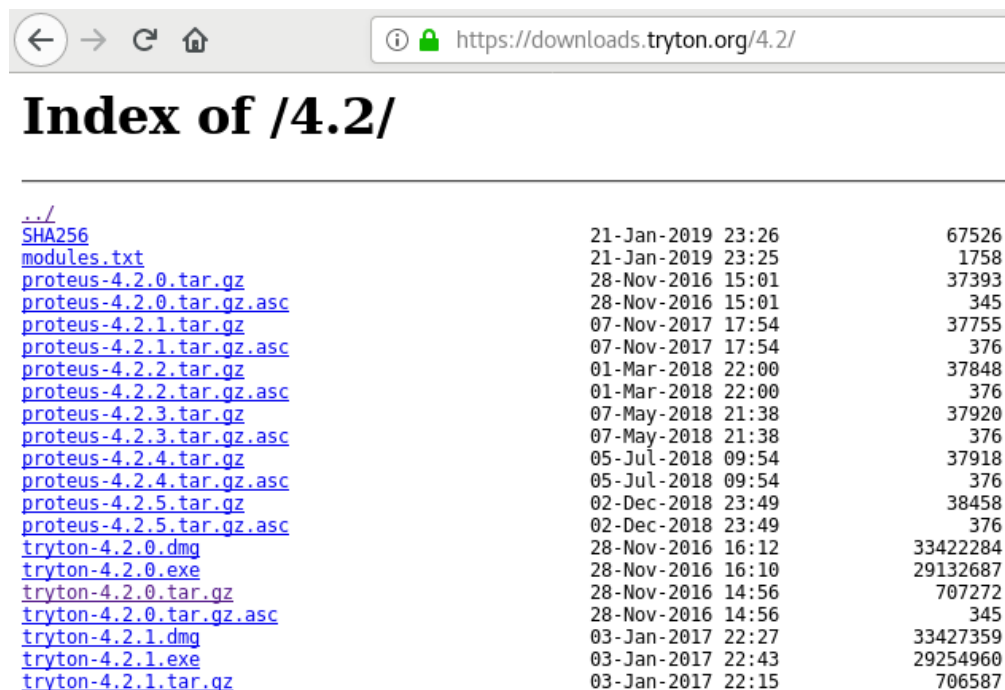


Figura 59. Arrancamos el servidor

## Descargar el cliente tryton desde la página oficial

<https://downloads.tryton.org/4.2/>

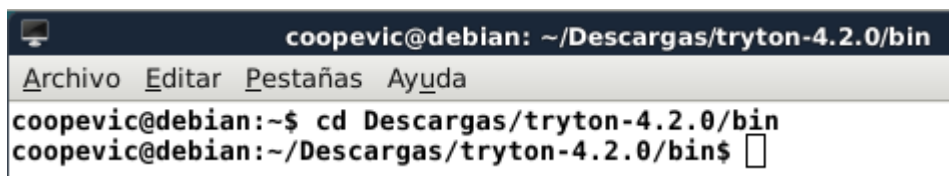


../	21-Jan-2019 23:26	67526
SHA256	21-Jan-2019 23:25	1758
modules.txt	28-Nov-2016 15:01	37393
proteus-4.2.0.tar.gz	28-Nov-2016 15:01	345
proteus-4.2.0.tar.gz.asc	07-Nov-2017 17:54	37755
proteus-4.2.1.tar.gz	07-Nov-2017 17:54	376
proteus-4.2.1.tar.gz.asc	01-Mar-2018 22:00	37848
proteus-4.2.2.tar.gz	01-Mar-2018 22:00	376
proteus-4.2.2.tar.gz.asc	07-May-2018 21:38	37920
proteus-4.2.3.tar.gz	07-May-2018 21:38	376
proteus-4.2.3.tar.gz.asc	05-Jul-2018 09:54	37918
proteus-4.2.4.tar.gz	05-Jul-2018 09:54	376
proteus-4.2.4.tar.gz.asc	02-Dec-2018 23:49	38458
proteus-4.2.5.tar.gz	02-Dec-2018 23:49	376
proteus-4.2.5.tar.gz.asc	28-Nov-2016 16:12	33422284
tryton-4.2.0.dmg	28-Nov-2016 16:10	29132687
tryton-4.2.0.exe	28-Nov-2016 14:56	707272
tryton-4.2.0.tar.gz	28-Nov-2016 14:56	345
tryton-4.2.0.tar.gz.asc	03-Jan-2017 22:27	33427359
tryton-4.2.1.dmg	03-Jan-2017 22:43	29254960
tryton-4.2.1.exe	03-Jan-2017 22:15	706587
tryton-4.2.1.tar.gz		

Figura 60. Descargamos el cliente tryton desde la página oficial

## Si lo dejamos en la carpeta descargas

`cd Descargas/tryton-4.2.0/bin`



```
coopevic@debian: ~/Descargas/tryton-4.2.0/bin
Archivo Editar Pestañas Ayuda
coopevic@debian:~$ cd Descargas/tryton-4.2.0/bin
coopevic@debian:~/Descargas/tryton-4.2.0/bin$
```

Figura 61. Accedemos a la carpeta donde se encuentra el cliente tryton

## Habilitar los permisos de ejecución

`sudo chmod 777 -R tryton`

`sudo ./tryton`

## Inicialización de Tryton

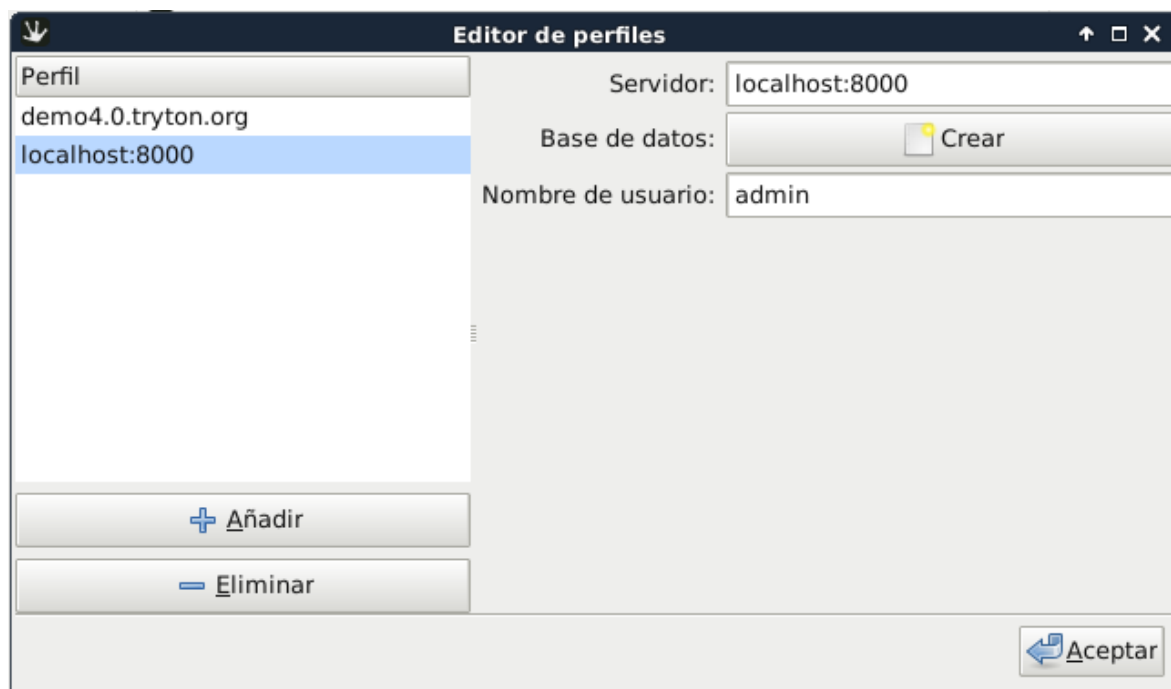
Inicialización de Tryton 4.0.



The screenshot shows the 'Usuario' (User) window in Tryton. It features the Tryton logo and a moon image. Below the header, there is a 'Perfil:' dropdown menu and an 'Administrar perfiles' button. A section titled 'Información del Servidor / Base de datos' contains fields for 'Servidor:' (localhost:8000), 'Base de datos:' (tryton), 'Nombre de usuario:' (admin), and 'Contraseña:'. At the bottom right, there are 'Cancelar' and 'Conectar' buttons.

## Editamos los perfiles

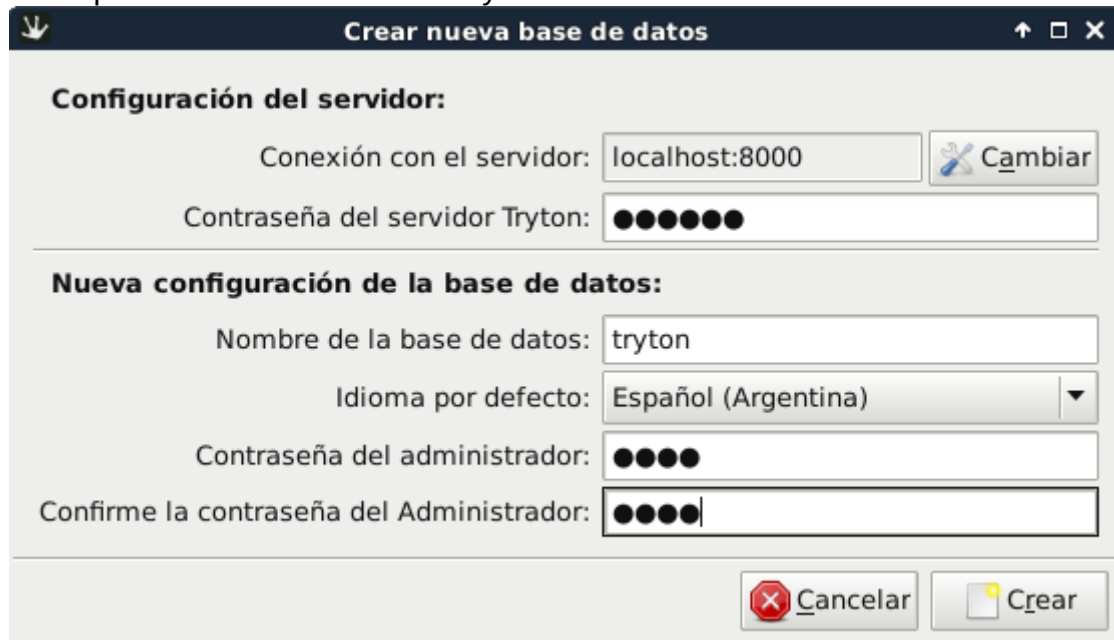
Con la versión de Tryton 4.0, se podía crear la base de datos desde Tryton



The screenshot shows the 'Editor de perfiles' (Profile Editor) window. On the left, a list of profiles includes 'demo4.0.tryton.org' and 'localhost:8000', with the latter selected. Below the list are 'Añadir' (Add) and 'Eliminar' (Remove) buttons. On the right, the 'Servidor:' field is set to 'localhost:8000', the 'Base de datos:' field has a 'Crear' (Create) button, and the 'Nombre de usuario:' field is set to 'admin'. An 'Aceptar' (Accept) button is at the bottom right.

## Creamos la base de datos

Esto era posible con la versión de Tryton 4.0



The screenshot shows a window titled "Crear nueva base de datos" (Create new database). It is divided into two main sections: "Configuración del servidor:" (Server configuration) and "Nueva configuración de la base de datos:" (New database configuration). In the server configuration section, there is a text field for "Conexión con el servidor:" (Connect to server) containing "localhost:8000" and a "Cambiar" (Change) button with a wrench icon. Below it is a password field for "Contraseña del servidor Tryton:" (Tryton server password) filled with dots. The database configuration section includes a text field for "Nombre de la base de datos:" (Database name) containing "tryton", a dropdown menu for "Idioma por defecto:" (Default language) set to "Español (Argentina)", a password field for "Contraseña del administrador:" (Administrator password) with four dots, and a confirmation field for "Confirme la contraseña del Administrador:" (Confirm administrator password) with four dots. At the bottom right are "Cancelar" (Cancel) and "Crear" (Create) buttons.

## Cargamos los datos del usuario

La contraseña de ingreso estaba en la misma pantalla de ingreso en Tryton 4.0



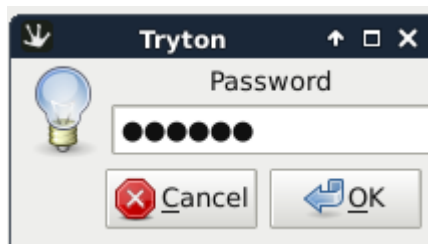
The screenshot shows a window titled "Usuario" (User). The top part features the Tryton logo and a background image of the moon. Below this is a "Perfil:" (Profile) dropdown menu set to "localhost:8000" and an "Administrar perfiles" (Manage profiles) button with a pencil icon. A section titled "▼ Información del Servidor / Base de datos" (Server / Database information) contains several fields: "Servidor:" (Server) with "localhost:8000", "Base de datos:" (Database) with "tryton", "Nombre de usuario:" (Username) with "admin", and "Contraseña:" (Password) with four dots. At the bottom right are "Cancelar" (Cancel) and "Conectar" (Connect) buttons.

## Inicialización de Tryton

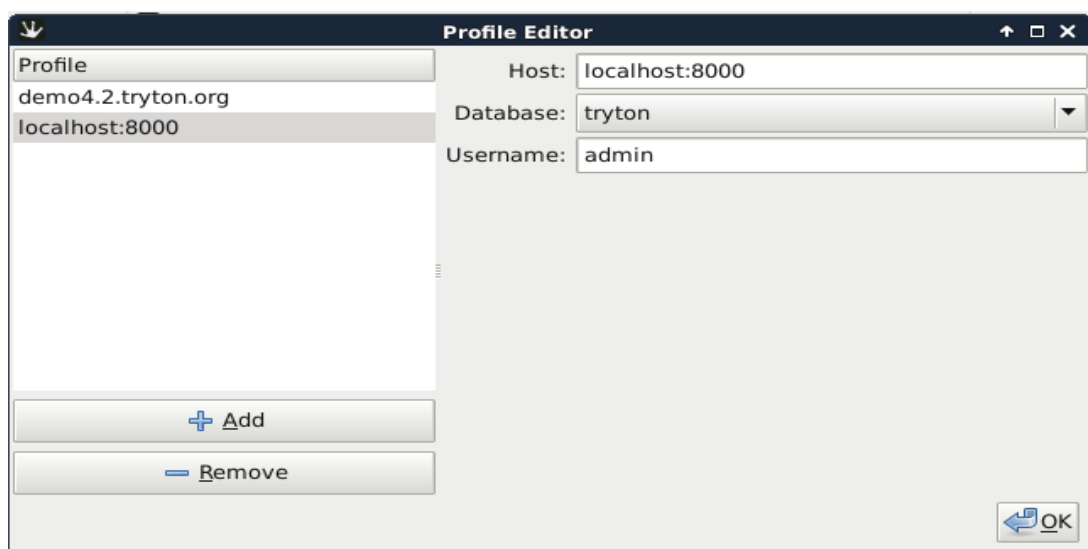
Inicialización de Tryton 4.2.



La contraseña de ingreso está en una ventana separada en Tryton 4.2

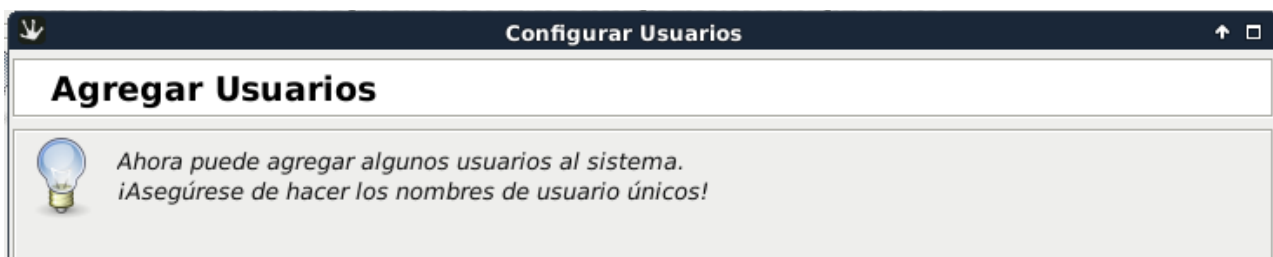
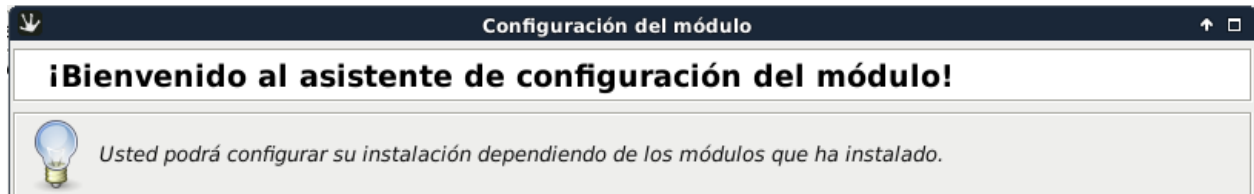


Con la versión de Tryton 4.2, la base de datos ya tiene que estar creada con anterioridad con Postgres.



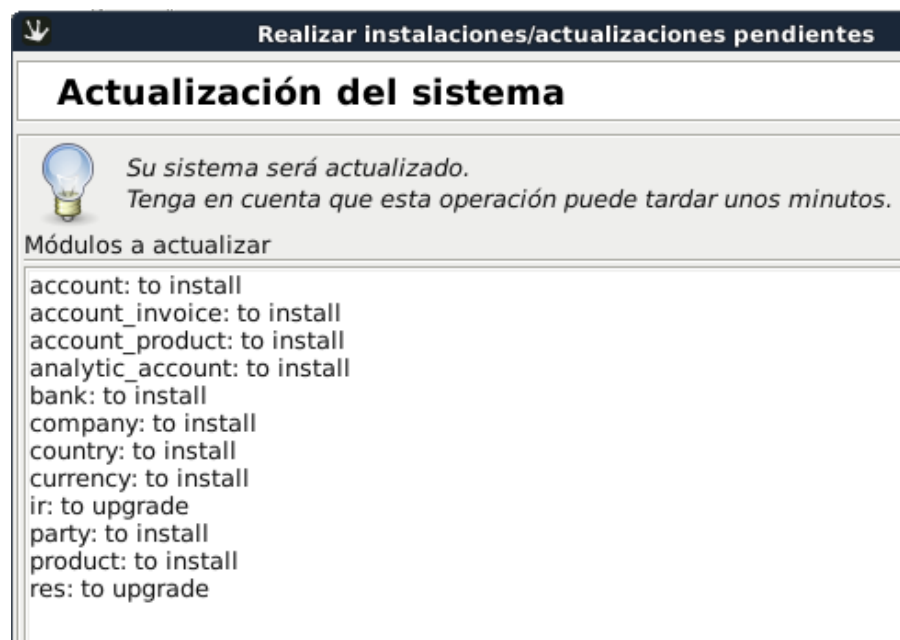
## Configuración de módulos y usuarios

---

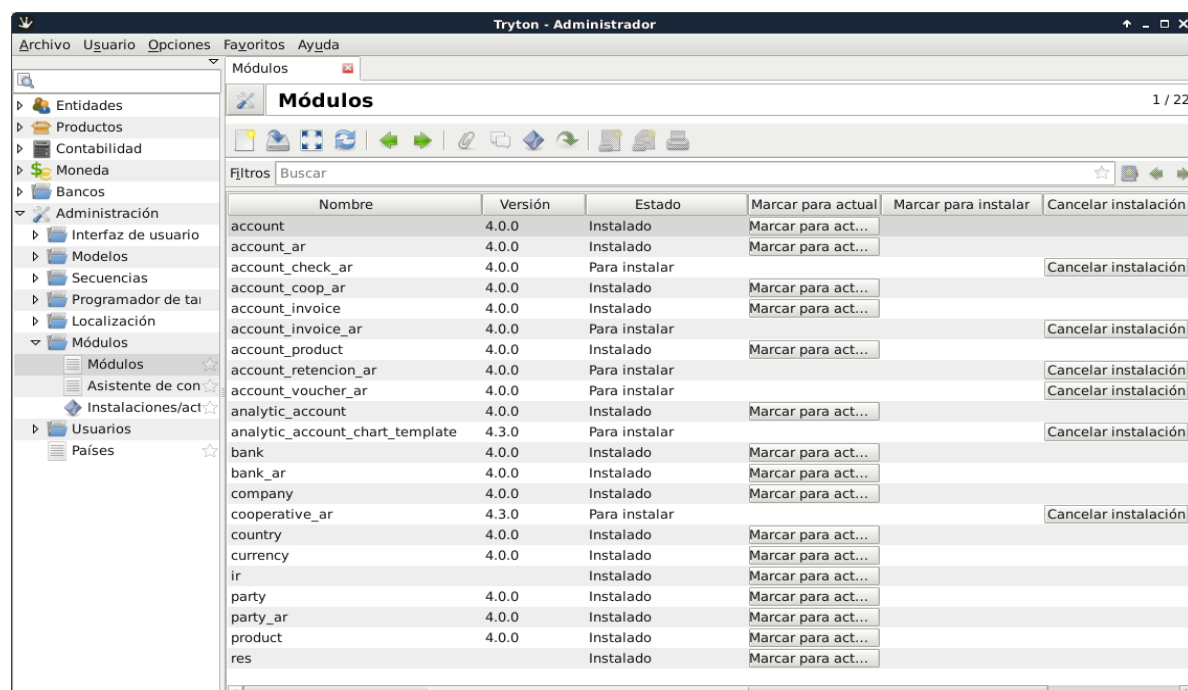


## Actualización del sistema

---



## Administración de Módulos



## Módulo Gestión Cooperativa

El Módulo cooperative\_ar contiene una serie de módulos para llevar adelante la gestión de una Cooperativa de Trabajo:

- Socios
- Reuniones
- Sanciones
- Vacaciones / licencias
- Recibos
- Balance Social Cooperativo

## Socios

Este módulo permite la carga de datos útiles del socio. Primero que nada el socio debe ser creado como Entidad, dado que ahí se cargarán los datos contables necesarios para que se opere en el sistema. Desde el módulo Socio se selecciona

al Socio (ya cargado como Entidad) y se cargan los datos complementarios de utilidad legal (legajo, estado, fecha de ingreso, etc.).

The screenshot shows the 'Partner' form in the Tryton ERP system. The form is titled 'Partner' and has a '1 / 1' indicator in the top right corner. It features a toolbar with various icons for file operations and a tabbed interface. The 'General' tab is selected, showing fields for personal and company information. The 'Datos Personales' section includes fields for Party, First Name, Last Name, Gender, DNI, Nationality, Marital Status, and Birthdate. The 'Incorporacion' section includes fields for Incorporation Date, Leaving Date, and Payed Quotes. The 'Legal Documents' section includes fields for Proposal Letter and Proof of tax registration. The 'Meeting date of incorporation' field is also present.

Este módulo permitirá entonces tener el legajo del socio dentro del sistema.

Este módulo está relacionado con el resto y es por eso que pueden visualizarse: las reuniones en las que el socio participó, sus recibos de Adelanto de Excedentes, las Sanciones y las vacaciones del mismo.

## Vacaciones

El módulo se puede utilizar creando nuevos registros desde el módulo o desde la pestaña de «vacaciones y licencias» que se visualiza en la edición de cada Socio.

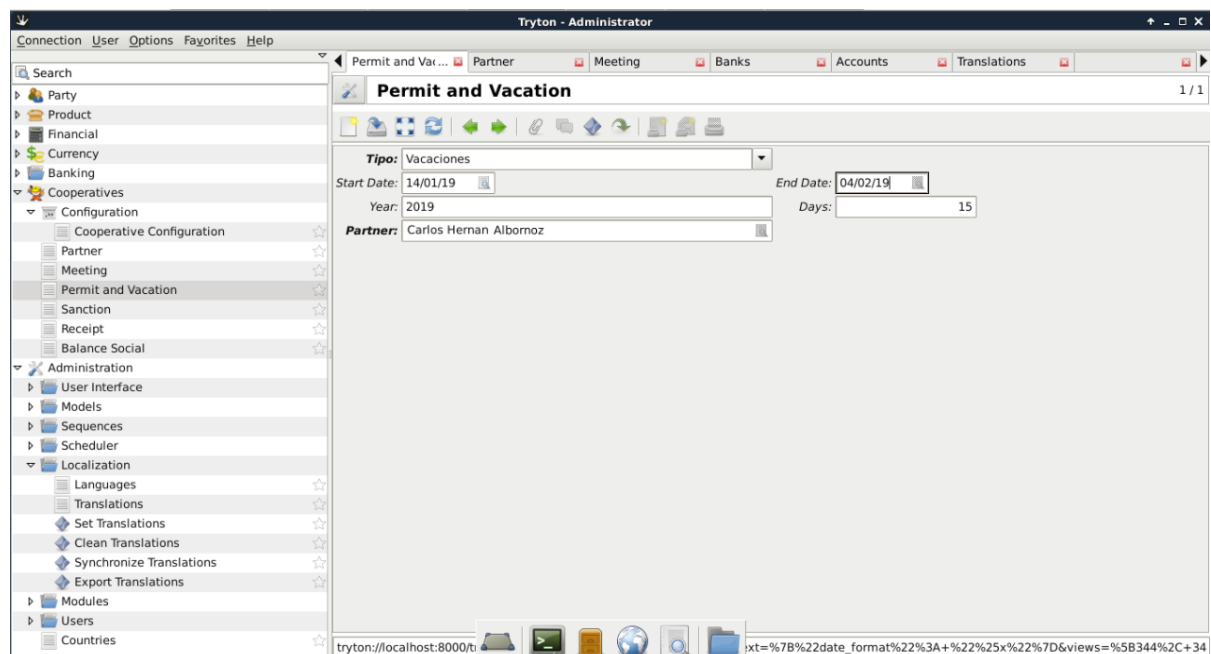


The screenshot shows a web-based form titled "Permit and Vacation" with a "1 / 1" indicator in the top right corner. The form contains the following fields:

- Tipo:** A dropdown menu set to "Vacaciones".
- Start Date:** A date field set to "14/01/19".
- End Date:** A date field set to "04/02/19".
- Year:** A text field set to "2019".
- Days:** A text field set to "15".
- Partner:** A text field set to "Carlos Hernan Alborno".

Below the form fields is a large, empty light-gray area.

Permite cargar los días que tiene asignados ese socio (Días de vacaciones) y luego permite cargar un registro por cada licencia o vacación que el socio se tome.



## Sanciones

Este módulo permite cargar sanciones a un socio con los Tipos que figuran en el Estatuto base de una Cooperativa de Trabajo: Llamado de Atención, Apercibimiento y Exclusión con el causante y el descargo presentado por el socio.

The screenshot shows the 'Sanction' module interface. At the top, there's a title bar with 'Sanction' and a page indicator '1 / 1'. Below the title bar is a toolbar with various icons for actions like create, edit, delete, and search. The main form area contains two input fields: 'Tipo:' with a dropdown menu set to 'Llamada de Atención', and 'Date:' with a date field set to '03/01/19'. Below these is a 'Partner:' field with the name 'Carlos Hernan Alborno' and a magnifying glass icon. Underneath the partner field is a 'Record' section with a text area containing the text 'Falta a reunión de asamblea sin aviso'. At the bottom of the form is a 'Response' section with a text area containing the text 'En el día dela fecha, el socio faltó a la reunión de asamblea por cuidado de familiar,'.

## Reuniones

Aquí se podrán cargar las Reuniones de Consejo y Asambleas que realiza la cooperativa, señalando los socios presentes y los temas tratados como área de texto o archivo adjunto (Tipos de Reunión: reunión de consejo, Asamblea Ordinaria o Extraordinaria).

**Meeting** 1 / 1

General Record

**Type:** Reunion de Consejo **Status:** Complete

**Start Date:** 09/01/19 **Start Time:** 17:30:00 **End Time:** 20:30:00

**Partner** Search

File	First Name	Last Name
1	Carlos Hernan	Albornoz
2	Carlos Alberto	Burgos
3	Jose Francisco	Ferrari
4	Gabriel	Balbi
5	Matias	Carrizo
6	Helvio	Bruera

Desde el socio se puede buscar una Reunión para señalar que el socio participó en la misma.

## Recibos

Permite crear el «Recibo de Anticipo de Retornos a Cuenta de Excedentes» que los socios de las cooperativas de trabajo reciben como prestación del trabajo que realizan.

**Recibos de Socio** 1 / 1

Partner:  Party:

Number:

Date:  Amount:

Description:  Journal:

Currency:

Receipt Other Info

Company:

Confirmed Move: ☐

Accounting Date:

Este módulo permite cargar el Socio, importe y la fecha del retiro y el diario sobre el que se imputará. Se genera un Recibo con un completo Workflow de trabajo. El sistema toma los datos de la Cooperativa y del socio de forma automática para generar la impresión del recibo de forma correcta. El recibo nace en borrador, pasa a estar inmutado una vez que está confirmado (tiene una secuencia propia de Recibos) y genera el asiento con la deuda y el mismo queda pendiente de pago.

Una vez que el mismo se paga queda terminado el proceso y se realiza el Asiento correspondiente. Toda la información contable es accesible desde el recibo.

**Recibos de Socio** 1 / 1

**Partner:** Jose Francisco Ferrari **Party:** Coop, de Trabajo COOPEVIC LTDA

**Number:** **Date:** 10/01/19 **Amount:** 2.000,00

**Description:** Retornos a cuenta de excedentes **Journal:** Cash

**Currency:** Argentine Peso

**Receipt** **Other Info**

**Company:** Coop, de Trabajo COOPEVIC LTDA

**Confirmed Move:**

**Accounting Date:**

El sistema está preparado para recibir la importación de datos de recibos para generar los mismos de forma automática. Se utiliza la importación del sistema y son necesarios los campos obligatorios del módulo (con documento del socio). Campos: Amount, Company, Currency, Journal, Date, Socio.

## Balance Social Cooperativo

El desarrollo de Balance Social Cooperativo le permite a la cooperativa cargar las actividades realizadas y valorizar actividades con impacto social que de otra manera quedan invisibilizadas en un Balance Contable. El reporte de Balance Social Cooperativo puede servir como información interna de la cooperativa o como insumo para la generación del Balance Social Cooperativo que de forma anual puede hacer la cooperativa. En tanto no existe un único modelo de Balance Social Cooperativo, el desarrollo realizado permite cargar y generar un informe con datos útiles sobre la gestión de la cooperativa. Para más información sobre Balance Social

Cooperativo puede verse la información disponible en la Alianza Cooperativa Internacional

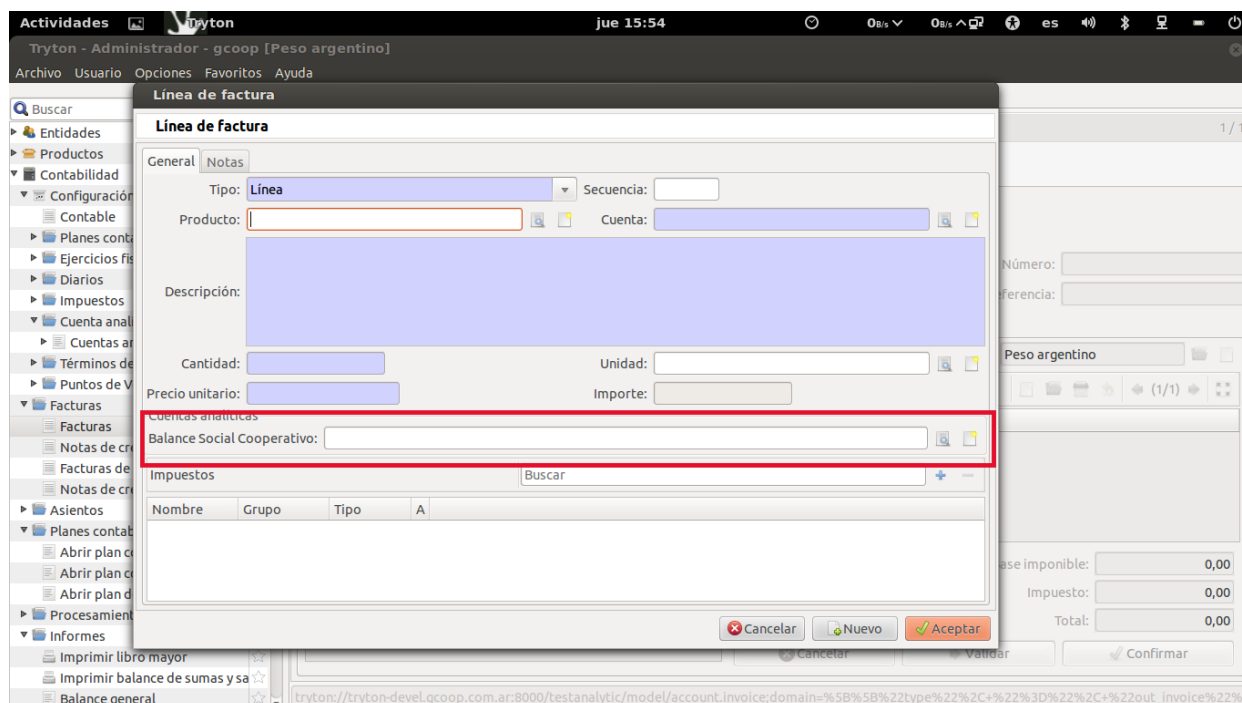
<http://www.aciamericas.coop/Balance-Social-Cooperativo-2394>

El desarrollo realizado se basa en toda la potencialidad y flexibilidad que brinda el concepto de Contabilidad Analítica propio del ERP. En este sentido lo que se agrega es un esqueleto para las cuentas analíticas bases de un Balance Social Cooperativo (los siete principios cooperativos como cuentas analíticas). Utilizar el concepto de cuentas

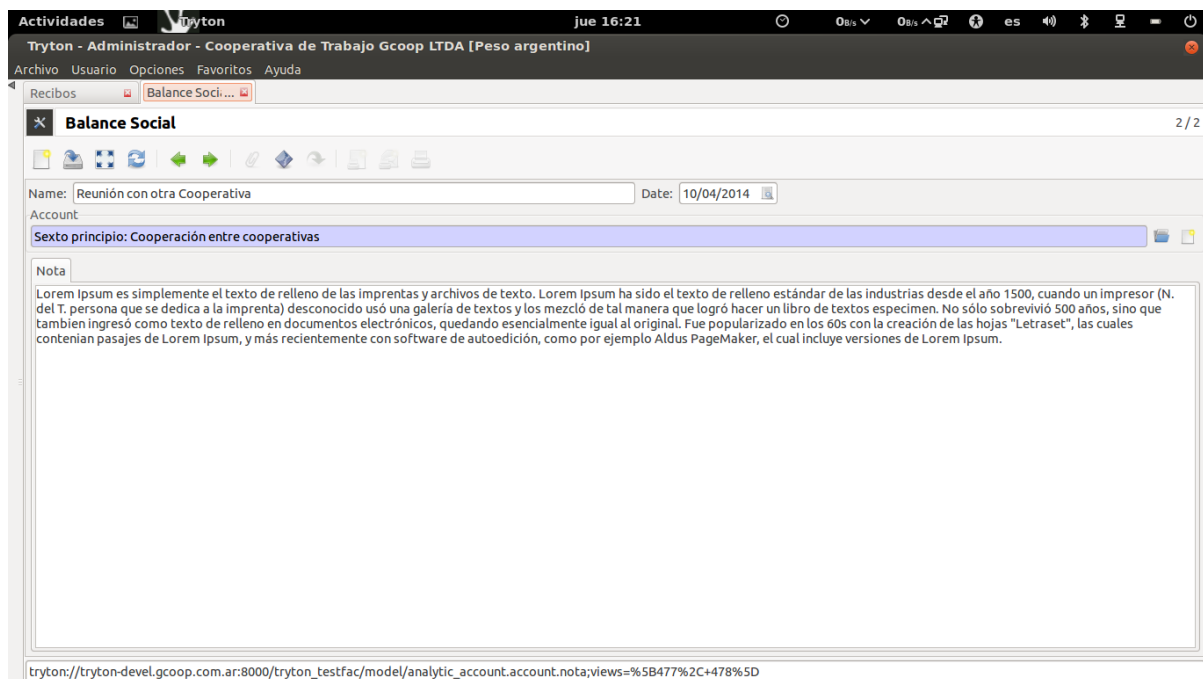
Nombre	Empresa	Tipo
1* - Balance Social Cooperativo	gcoop	Raíz
11 - Primer principio: Libre adhesión	gcoop	Normal
12 - Segundo principio: Control democrático de los socios	gcoop	Normal
13 - Tercer principio: Participación económica de los socios	gcoop	Normal
14 - Cuarto principio: Autonomía e independencia	gcoop	Normal
15 - Quinto principio: Educación, entrenamiento e información	gcoop	Normal
16 - Sexto principio: Cooperación entre cooperativas	gcoop	Normal
17 - Séptimo principio: Compromiso con la comunidad	gcoop	Normal

Los módulos Tryton: «Analytic Invoice», «Analytic Purchase» y «Analytic Sale» permiten entonces que toda compra o venta (o desde Facturas) puedan ser imputadas no solo a las cuentas contables sino también a las cuentas analíticas del

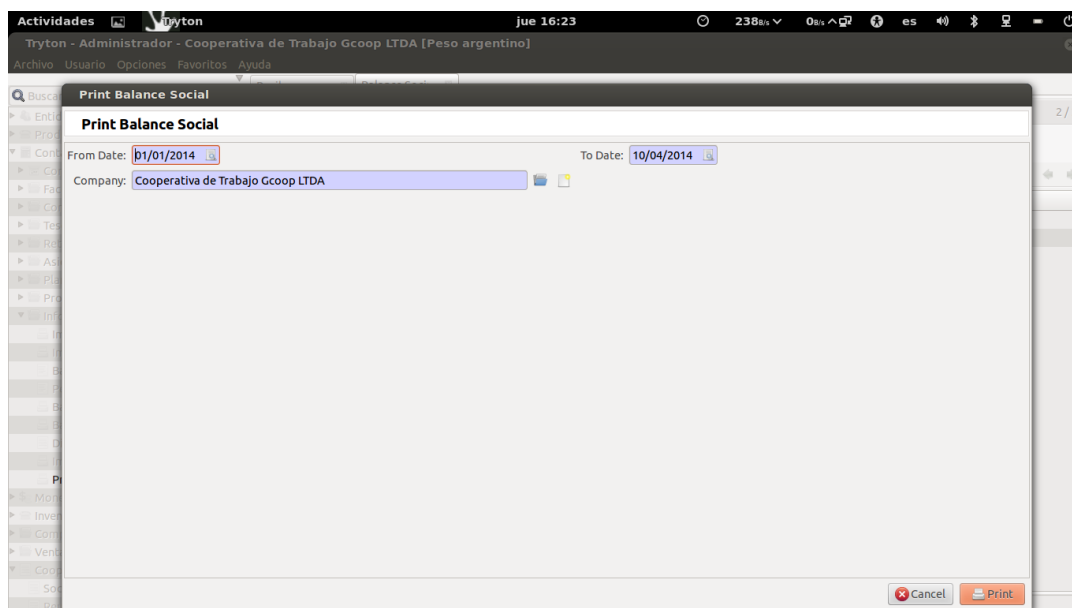
## Balance Social Cooperativo.



Por ejemplo una compra realizada a una cooperativa puede imputarse al principio 6 «Cooperación entre cooperativas». Esto nos permite ir registrando y valorizar en la gestión cotidiana el trabajo social de la cooperativa. Por otro lado se desarrolló un módulo que permite cargar Notas que se asocian a las líneas analíticas del Balance Social Cooperativo. Esto permite llevar una memoria narrativa de las actividades realizadas por la cooperativa.



Toda la información que se genera tanto desde la imputación analítica al Balance Social Cooperativo de los movimientos contables como las notas cargadas al sistema forman parte del Informe Balance Social Cooperativo que se exporta en formato editable.





La exportación permite seleccionar el rango de fechas deseado y la empresa (es multiempresa) sobre la que se quiere generar el informe. El Balance Social Cooperativo contiene además una serie de datos útiles sobre la gestión de la cooperativa durante el período seleccionado. Se exporta el listado de Socios, la composición por Sexo, el listado de altas y bajas, la cantidad de reuniones realizadas y la presencia de los socios en las mismas.

## Publicación en gitlab

---

En Gitlab <https://gitlab.com/Hertux/Trabajo-Final>, no solamente se publicará esta documentación en formato ODT y PDF, sino también los archivos de configuración y requerimientos, utilizados durante la instalación de Tryton y que son los siguientes:

- **.bashrc** (archivo oculto)
- **install-trytonar.sh** (script)
- **requirements.txt** (archivo de requerimiento)
- **trytond.conf** (archivo de configuración)
- **tryton-4.2.0.tar.gz** (archivo comprimido de tryton)

## Conclusión

---

Una de las conclusiones es la escasa documentación técnica que existe de tryton. Es muy difícil encontrar una guía "paso a paso" amigable para novatos. La instalación es muy compleja y requiere de muchos pasos para realizarla y se debe tener en cuenta que al cambiar las versiones de Tryton, el proceso de instalación puede variar un poco. Lo bueno es que existe una gran comunidad dispuesta a colaborar, ya sea con material de consulta o brindando ayuda a través de los foros de discusión o a través del mail.

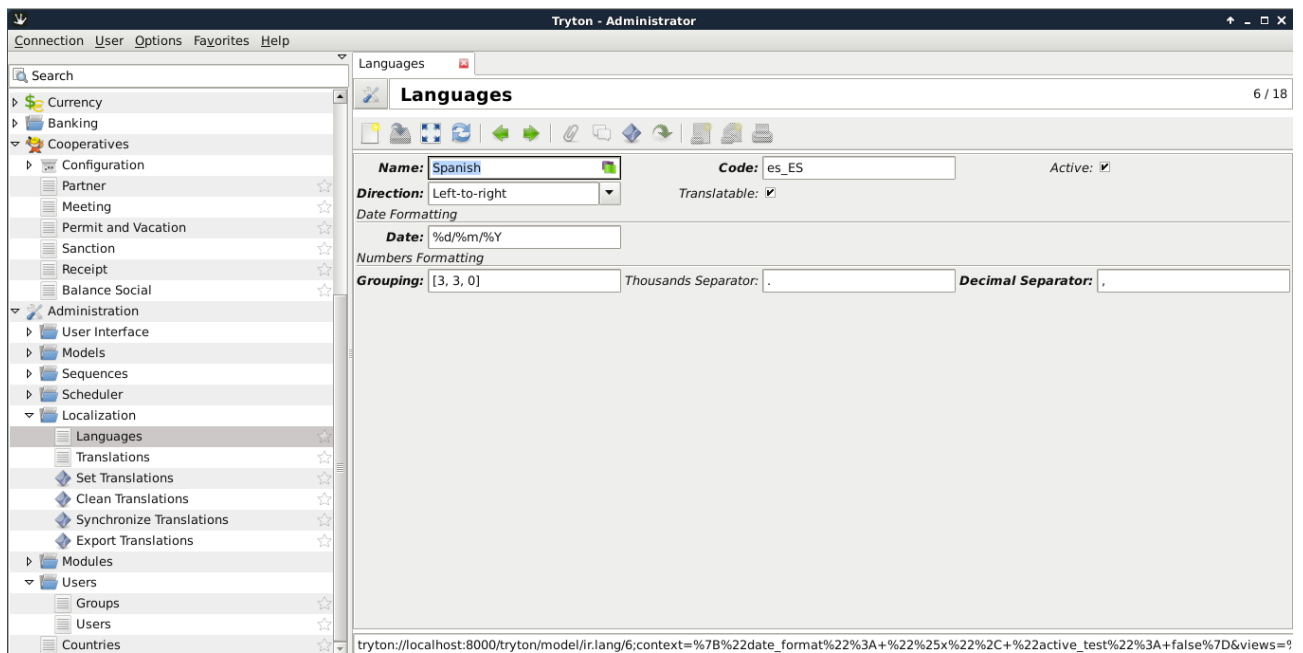
Hay empresas como Thymbra que ofrecen cursos a otras empresas para la instalación de esta plataforma ERP, pero creo que sería necesario también que se brindaran cursos por ejemplo a través de videos tutoriales, ya que todos los videos existentes solo explican como es el uso de esta plataforma y no como se realiza la instalación.

Cabe mencionar además, que durante la instalación del módulo de cooperativas, cuando quería arrancar el servidor Tryton, me daba el siguiente error:

```
Traceback (most recent call last):
  File "/home/coopevic/.virtualenvs/tryton/bin/trytond-admin", line 21, in <module>
    admin.run(options)
  File "/home/coopevic/.virtualenvs/tryton/local/lib/python2.7/site-packages/trytond/admin.py", line 48, in run
    Pool(db_name).init(update=options.update, lang=lang)
  File "/home/coopevic/.virtualenvs/tryton/local/lib/python2.7/site-packages/trytond/pool.py", line 144, in init
    self.start()
  File "/home/coopevic/.virtualenvs/tryton/local/lib/python2.7/site-packages/trytond/pool.py", line 97, in start
    register_classes()
  File "/home/coopevic/.virtualenvs/tryton/local/lib/python2.7/site-packages/trytond/modules/__init__.py", line 362, in register_classes
    mod_file, pathname, description)
  File "/home/coopevic/.virtualenvs/tryton/local/lib/python2.7/site-packages/trytond/modules/cooperative_ar/__init__.py", line 8, in <module>
    from .account import *
ImportError: No module named account
(tryton) coopevic@debian:~/virtualenvs/tryton/lib/python2.7/site-packages/trytond/modules$
```

Reporté el error via e-mail y gracias a la buena predisposición y a la rapidez de la respuesta y a la solución del problema por parte de Luciano Rossi, socio de la cooperativa GCOOP, pude solucionar el problema.

Otro dato importante es el cambio de lenguaje. En esta versión de Tryton se debe utilizar el lenguaje "Español" y no "Español (Latinoamericano)". Sucede que desde la v4.2 hubo un cambio en los lenguajes y el latino quedo con algunas partes en ingles y otras en castellano. Igualmente al cambiar al lenguaje "Español" no se produjo el cambio que esperaba. Aparentemente este problema se corrigió a partir de la versión 4.8 de Tryton.



## Bibliografía

---

- <http://tryton-ar.readthedocs.io/es/latest/introduccion.html>
- <https://www.gcoop.coop/gestion-cooperativa-con-tryton-erp>
- <https://www.gcoop.coop/gestion-cooperativa-con-tryton-erp>
- <https://www.argentina.gob.ar/noticias/el-software-libre-como-herramienta>
- [www.tryton.org.ar/](http://www.tryton.org.ar/)
- <https://github.com/tryton-ar/>
- <https://groups.google.com/d/forum/tryton-ar>