

README

IMPLEMENTACIONES FUTURAS

- [] Las funciones aplicar retornan boolean, en funcion aceptar hacer if(aplicar()) para que si hay fallo al aplicar no se haga el self.accept()
- ☒ al insertar pone el dia de la semana a hoy
- ☐ Modificar asistente inicial, hacer un qtext disable que muestre la ruta por defecto, y si lo habilitas pues poder modificarla, cuando lo habras si ya apunta hacia una ruta que la muestre para que vea que esta bien
- ☐ en descarga automatica poner con self.MuestraNotificaciones y que no sea variable global
- ☐ Cada dia un mensaje de las series para ese dia
- ☐ En actualizar un boton que ponga directamente el dia de hoy al dia que hubiese puesto y revisar que al insertar aparezca el dia en el que se inserta
- ☐ Arreglar problema de importar librerías, hago un cambio y no funciona hasta que vuelvo a abrir la aplicación
- ☐ Hacer botón aplicar y guardar, uno aplica y muestro todo correcto y otro aplica y cierra (SIGUIENTE)
- ☐ En todos los botones aceptar/aplicar mostrar mensaje
- ☐ Repasar library.zip en linux
- ☐ Al actualizat/insertar relacionar los campos estado y siguiendo, (si le doy a finalizar que el estado sea finalizar)

Para qué es este repositorio:

- Esto es un programa para la gestión de todas las series que sigues; sus principales funcionalidades son seguir capítulo a capítulo las series y descargar series automáticamente (torrent) cuando sale un capitulo nuevo.
- Está programado con Python 3.4.4 y PyQt4.
- Versión 4.1.0

Funcionalidades

Series

- **Series Activas:** Muestra todas las series activas con un botón de sumar o restar capítulos; para que se apliquen los cambios en los capítulos tienes que darle a aplicar para que se ejecute la query en la base de datos. (*Ctrl + A*)
- **Modificar Series:** Te permite modificar las series; para cambios generales (acaba/empieza temporada) se puede usar *modificación en masa*, pero para cambios más específicos se usa la modificación individual.
 - **Modificación Masiva:** Muestra las series para hacer modificaciones; te permite cambiar los estados de muchas series. Para que se apliquen los cambios hay que darle a aplicar.
 - **Modificación individual:** Busca una serie específica en la base de datos y te abre la ventana de modificación de la serie. (*Ctrl + F*)
- **Insertar Serie:** Abre una ventana para meter una nueva serie en la base de datos. (*Ctrl + I*)
- **Revisar Estados:** Revisa los estados de las series (si empiezan la temporada, acaban la temporada, si acaban la serie, cuándo sale una serie); le tienes que dar a añadir y para que se ejecute la query y darle a aplicar. (*Ctrl + E*)
- **Salir:** Cierra el programa. (*Ctrl + X*)

Herramientas

- **Actualizar db de Imdb:** En proceso de implementar Actualiza la base de datos de las series con los valores sacados de [Imdb](#)
- **Descarga Automática:** Comprueba todas las series que sigo; en caso de que haya salido algún capítulo nuevo, se descarga el torrent en la ruta indicada y envía una notificación con el servicio que esté habilitado. El programa de torrent que uses es el que se debe de encargar de coger automáticamente el torrent de esa ruta y empezar a descargarlo; en caso de que no haya Internet, se cierra.
 - **Newpct1:** Comprueba las series en español (*Ctrl + N*)
 - **ShowRss:** Comprueba las series en V.O.S.E (*Ctrl + S*)

Opciones

- **Preferencias:** Para cambiar la configuración por defecto del programa puedes tener distintas configuraciones. Puedes añadir más configuraciones

cambiando el id a otro. (*Ctrl + P*)

- **Notificaciones:** Para habilitar o deshabilitar los servicios de notificaciones cuando nos descargamos una serie (*Ctrl + K*)
 - Telegram: poner el id de la cuenta a la que enviar un mensaje.
 - Pushbullet: poner la api de tu cuenta.
 - Email: poner el mail al que enviar los mensajes.
 - Hangouts: actualmente no funciona.
- **Vaciar log:** Borra el fichero de log que elegimos de log. Se usa para reiniciar las descargas automáticas y hacer que vuelva a revisar todo el listado del feed (no solo desde la ultima entrada en la que se quedó).
 - **Newpct1:** Vacía el log de Newpct1 (*Ctrl + 1*)
 - **ShowRss:** Vacía el log de Showrss (*Ctrl + 2*)
 - **Descargas:** Vacía el log de Descargas (donde se guardan todas las series que se descargan) (*Ctrl + 3*)
 - **Todos:** Vacía todos los logs (*Ctrl + 4*)
- **Id de opcion: En proceso de implementar** Para seleccionar el id de la configuración que está activo de forma rápida; de momento se hace a través de las preferencias.

Ayuda

- **Acerda De:** (*Ctrl + H*)

Configuración de la base de datos

```
BEGIN TRANSACTION;
CREATE TABLE "Series" (
  `ID` INTEGER NOT NULL PRIMARY KEY AUTOINCREMENT UNIQUE,
  `Nombre` TEXT NOT NULL UNIQUE,
  `Temporada` INTEGER DEFAULT 1,
  `Capitulo` INTEGER DEFAULT 1,
  `Siguiendo` TEXT DEFAULT 'Si',
  `VOSE` TEXT DEFAULT 'No',
  `Acabada` TEXT DEFAULT 'No',
  `Dia` TEXT DEFAULT 'None',
  `Estado` TEXT DEFAULT 'Activa',
  `imdb_id` TEXT UNIQUE,
  `imdb_Temporada` TEXT,
  `imdb_Capitulos` TEXT,
  `imdb_Finaliza` TEXT,
  `imdb_seguir` INTEGER DEFAULT 'Si'
);
CREATE TABLE "Notificaciones" (
  `ID` INTEGER NOT NULL PRIMARY KEY AUTOINCREMENT UNIQUE,
  `Nombre` TEXT UNIQUE,
  `API` TEXT,
  `Activo` INTEGER DEFAULT 0
);
CREATE TABLE `ID_Estados` (
  `ID` INTEGER NOT NULL PRIMARY KEY AUTOINCREMENT UNIQUE,
  `Estados` TEXT UNIQUE
);
CREATE TABLE "Configuraciones" (
  `id` INTEGER NOT NULL PRIMARY KEY AUTOINCREMENT,
  `BaseDeDatos` TEXT,
  `UrlFeedNewpct` TEXT,
  `UrlFeedShowrss` TEXT,
  `RutaDescargas` TEXT,
  `FicheroFeedNewpct` TEXT,
  `FicheroFeedShowrss` TEXT,
  `FicheroDescargas` INTEGER
);
COMMIT;
```

Requerimientos

Para compilar el código fuente, se requieren algunas bibliotecas externas. Las librerías que he usado con sus respectivas versiones a dia de hoy (10/08/2016) son:

* Python 3.4

```
* PyQt4 (4.11)
* requests (2.7.0)
* feedparser (5.2.1)
* wget (2.2)
* BeautifulSoup (3.2.1)
* mechanize (0.2.5)
* imdbpie (4.0.2)
* pushbullet.py (0.8.1)
* mailer (0.8.1)
```

Instalación en Windows:

1. Tener instalado python 3.4 y pip3 en el sistema, y que ambos esten en el path de windows.
2. Instalar las librerías necesarias para python 3.

```
pip3 install -r requirements.txt
```

3. Instalar PyQt4 para python 3.

- [url 32 bits](#) (El que uso actualmente)
- [url 64 bits](#)

Instalación en Linux

1. Tener instalado python 3.4 y pip3 en el sistema, esta testeado en *Debian 7* y *Fedora 24*, el resto de distribuciones no estan testeadas.

Debian:

```
sudo apt-get install python3-pip python3-pyqt4 python3-pyqt4.qtsql libqt4-sql-sqlite python-qt4-sql libqt4-sql-sqlite
```

Fedora:

```
sudo dnf install python3-pip python3-PyQt4.x86_64 PyQt4-devel
```

2. Instalar las librerías necesarias para python 3.

```
pip3 install -r requirements.txt
```

Comandos restantes

librerías para generar ficheros #Testear si hace falta en fedora, sino añadirla directamente a la lista de comando de debian

```
sudo apt-get install pyqt4-dev-tools
```

Para generar el fichero de imágenes:

```
pyrcc4 -py3 fatcow.qrc -o fatcow_rc.py
```

para generar los ficheros ui

```
pyuic4 fichero.ui -o fichero.py
```

License

GNU GENERAL PUBLIC LICENSE

Version 3, 29 June 2007

Pablo Rocamora

pablojoserocamora@gmail.com
