

# Práctica 1 - Interfaces Gráficas de Usuario con Gtk+ y Python

Interfaces Persona Máquina

18/9/2017 – 13/10/2018

## Objetivos

Aplicar los conocimientos adquiridos sobre el desarrollo de interfaces gráficas de usuario durante el desarrollo de una aplicación gráfica de escritorio. Dicha aplicación se implementará utilizando el lenguaje de programación Python y la librería gráfica Gtk+.

## 1. Descripción del trabajo

Al llegar a tu nuevo trabajo te encuentras una aplicación desarrollada por el equipo que acaban de despedir. La aplicación es el mítico gestor de una lista de tareas donde puedes añadir tareas pendientes de realizar, indicar su fecha de finalización, marcar las que ya se han realizado, etc.

Tu jefe te ha encargado que rehagas la aplicación porque se ha dado cuenta que presenta muchos fallos. El diseño de la interface es muy pobre y presenta muchas carencias, y el diseño software es, simplemente, inexistente.

Como tu jefe confía en ti y sabe que vas a realizar un buen trabajo, también te ha encargado que después de rehacer la aplicación, le añadas las funcionalidades que están pendientes.

## 2. Cómo realizar la práctica

En los siguientes apartados se describen las tareas que debes llevar a cabo. Si deseas que tu trabajo reciba una valoración positiva es necesario que te ciñas a las siguientes indicaciones:

- Las tareas deben realizarse en el orden en que se presentan en el enunciado de la práctica.

Por supuesto, es posible un desarrollo iterativo en el que se corrigen errores que no se hayan detectado durante el desarrollo de tareas anteriores.

Es cierto que el orden propuesto no encaja con una metodología actual de ingeniería del software, sin embargo, no debemos olvidar el objetivo de la práctica, cada tarea esta pensada para reforzar el aprendizaje de aspectos concretos de la materia y, para que este aprendizaje sea lo más efectivo posible, deben realizarse en el orden estipulado.

- No se evaluará ningún material que no esté en el repositorio. Esto incluye el código fuente, pero también la documentación, diagramas, datos, ... Por tanto, no olvides incluir en el repositorio todos los resultados de cada una de las tareas.

Tampoco olvides que estás trabajando con un sistema de control de versiones distribuido. Trabaja lo más posible contra tu repositorio local, y usa los repositorios para sincronizar tu trabajo con el de tu compañero de prácticas. Y no tengas miedo de hacer commits, no racanees, el sistema se encarga de calcular y almacenar sólo las diferencias entre una versión y la siguiente para optimizar el espacio.

- Las ramas de versiones del repositorio se usarán para tomar decisiones a la hora de evaluar la práctica. A continuación te mostramos unos ejemplos en los que la práctica se considera no apta:

- El orden de las versiones no es el indicado para las tareas en el enunciado de la práctica.
- Todas las versiones (commits) se realizaron el último día.

- El repositorio de referencia para la evaluación de la práctica es el que se encuentra en los servidores proporcionados por el Cecafi. Asegúrate de que su contenido este completo.

Recuerda que en cualquier momento puedes clonar el repositorio para comprobar su contenido. Por ejemplo, una vez clonado `git reset --hard sprint1` nos “resetea” nuestra copia de trabajo hasta la versión que habíamos etiquetado como `sprint1`.

- EL proyecto que contiene el repositorio en el servidor del Cecafi se tiene que llamar *exactamente*: `ipm-p1`.

## 2.1. Requisitos no funcionales

- La implementación se realizará con python y GTK+. Ambas en su versión 3.

## 3. Descripción de la aplicación

La aplicación es la típica *TODO List* que gestiona una lista de tareas pendientes y realizadas. Cada tarea viene dada por una descripción textual,

una fecha límite y una variable que indica si se ha realizado o no. Los casos de uso soportados son los siguientes:

- Añadir una tarea a la lista de tareas
- Eliminar una tarea de la lista de tareas
- Editar los datos de una tarea
- Marcar/desmarcar una tarea como hecha
- Ordenar las tareas

## 4. Tareas a realizar

Los siguientes apartados describen las *tareas* que debes realizar según la planificación establecida.

### 4.1. Reingeniería. Documentar

A los defectos de la interface actual se añade el hecho de que no está documentada, así que comenzaremos por corregir esta carencia.

1. Documenta la interface actual. Documenta las ventanas y diálogos existentes mediante *wireframes*, y el flujo de todas las acciones posibles. Se exhaustivo, no te limites al comportamiento esperado de la interface, documenta también las situaciones de error aunque no se traten en la implementación inicial. Por ejemplo: pulso el botón 'Eliminar' sin haber seleccionado una tarea.

A la última versión que se corresponda con la realización de este apartado asígnale la etiqueta **task1**.

### 4.2. Reingeniería. Diseño software

Dado que el diseño software es inexistente, pasamos a corregirlo en este apartado para facilitar la realización de las siguientes tareas.

1. Estudia el código existente. Difícilmente podrás hacer ningún cambio si no lo entiendes.
2. Realiza un diseño software de la aplicación aplicando el patrón *Model View Controller*.

A estas alturas de la carrera no hace falta que te lo diga, pero ya tú sabes, usa diagramas UML para documentar el diseño.

3. Reimplementa la aplicación siguiendo el diseño que acabas de realizar.

A la última versión que se corresponda con la realización de este apartado asígnale la etiqueta **task2**.

### 4.3. Reingeniería. Diseño e implementación

Llegados a este punto, procede a subsanar los fallos de la interface. Corrige el diseño de la misma e implementa los cambios.

Los principales aspectos a tratar son los siguientes:

- Gestionar los errores. Recuerda que ya te fijaste en ellos en la tarea del apartado 4.1.
- Mejora la eficiencia del uso de la interface. En la mayor parte de los casos, esto implicará reducir el número de pasos necesarios para realizar un caso de uso. Recuerda que tienes diagramas de flujo en la documentación de la interface donde este aspecto se ve claramente.
- Dado que la interface se implementa con la librería Gtk+, su entorno de ejecución natural es el escritorio Gnome. Por tanto, debes adecuar el diseño de la interface a las Gnome HIG<sup>1</sup>.

Crea un documento de texto con los apartados de cada una de las guías que aplica a la interface de la aplicación. En los casos en que sea necesario cambia el diseño para que se ajuste a las guías.

Durante la realización de este apartado, no te sorprendas por el hecho de que la intersección de los items propuestos no está vacía. Dicho de otra forma, para cumplir alguna de las guías tendrás que mejorar algunos aspectos del uso de la interface o gestionar mejor los errores.

Respecto al control de versiones, dentro de este apartado *no hagas commits que incluyan más de un cambio en la interface*.

A la última versión que se corresponda con la realización de este apartado asígnale la etiqueta **task3**.

### 4.4. Nuevo caso de uso

En este paso, debes añadir un nuevo caso de uso a la aplicación:

- Sincronizar la lista de tareas con un servidor remoto.

Para simplificar el desarrollo, el servidor remoto será simulado mediante una función que recibe una lista de tareas, como efecto colateral simula la

---

<sup>1</sup><https://developer.gnome.org/hig/stable/>

sincronización a través de un servicio de red y devuelve un valor indicando si la operación tuvo éxito o si se produjo algún error.

Dada la naturaleza de las operaciones de red, debes tener especial cuidado con las cuestiones relativas a la usabilidad como no interrumpir el funcionamiento de la interface o proporcionar el *feedback* adecuado.

A la última versión que se corresponda con la realización de este apartado asígnale la etiqueta **task4**.

#### 4.5. Internacionalización

Finalmente, la última tarea será internacionalizar la aplicación para que se ajuste automáticamente a la configuración de entorno local (*locale*) del usuario.

En concreto, se deben adaptar el idioma del texto de la interface y el formato de las fechas.

A la última versión que se corresponda con la realización de este apartado asígnale la etiqueta **task5**.

### 5. Evaluación de la práctica

Una vez terminada la práctica tienes que hacer una presentación de la misma al profesor para que la evalúe.

Para evaluar la práctica, el profesor seguirá los siguientes criterios:

- Si el repositorio no cumple con lo establecido en la sección 2, la práctica se evaluará como *no apta*, nota numérica 0.
- Si alguno de los integrantes del grupo, por su desconocimiento de la práctica, muestra no haber participado en la realización de la misma, esa persona recibirá una evaluación en la práctica de *no apta*.
- Cada una de las tareas que conforma la práctica se evaluarán de la siguiente manera:
  - **task1**
    - Hasta 0,75 puntos si se han documentado correctamente todas las ventanas y diálogos.
    - Hasta 0,5 puntos si se han documentado correctamente todos los flujos de las acciones cuando no hay errores.
    - Hasta 0,75 puntos si se han documentado correctamente todos los flujos de las acciones cuando hay errores.
  - **task2**
    - Hasta 0,75 puntos si el diseño sigue el estándar UML.

- Hasta 0,75 puntos si el diseño sigue el patrón MVC.
  - Hasta 0,5 puntos si la implementación se corresponde con el diseño.
  - -0,5 puntos si el diseño no es anterior a la implementación<sup>2</sup>.
- **task3**
  - Hasta 1 punto si la interface gestiona los errores.
  - Hasta 0,75 puntos si se ha mejorado la eficiencia de uso de la interface.
  - Hasta 0,75 si la interface sigue las Gnome HIG.
- **task4**
  - Hasta 2 puntos si la sincronización no bloquea la interface.
  - Hasta 0,5 puntos si se proporciona un feedback adecuado.
- **task5**
  - Hasta 0,5 puntos si los textos de la interface se adaptan al idioma de la usuaria.
  - Hasta 0,5 puntos si las fechas se adaptan al entorno local de la usuaria.
- El resultado de evaluar las tareas de la práctica constituye la nota provisional de la misma. A continuación, si procede, se aplican las penalizaciones por finalizar la práctica fuera de plazo.
  - Si la práctica se finalizó dentro del plazo establecido, esta nota se convierte en la nota final.
  - Si la práctica se finalizó fuera del plazo establecido:
    - Cuando finaliza con menos de un semana de retraso, la nota final es el mínimo entre 8,99 y la nota provisional.
    - Cuando finaliza con más de un semana de retraso, la nota final es el mínimo entre 6,99 y la nota provisional.

---

<sup>2</sup>Recuerda que estos aspectos se verifican con las versiones del repositorio.