



PONTIFICIA UNIVERSIDAD CATÓLICA DE CHILE
ESCUELA DE INGENIERÍA
DEPARTAMENTO DE CIENCIA DE LA COMPUTACIÓN
IIC2233 - PROGRAMACIÓN AVANZADA

Actividad 10

1º semestre 2018

24 de mayo

Interfaces gráficas

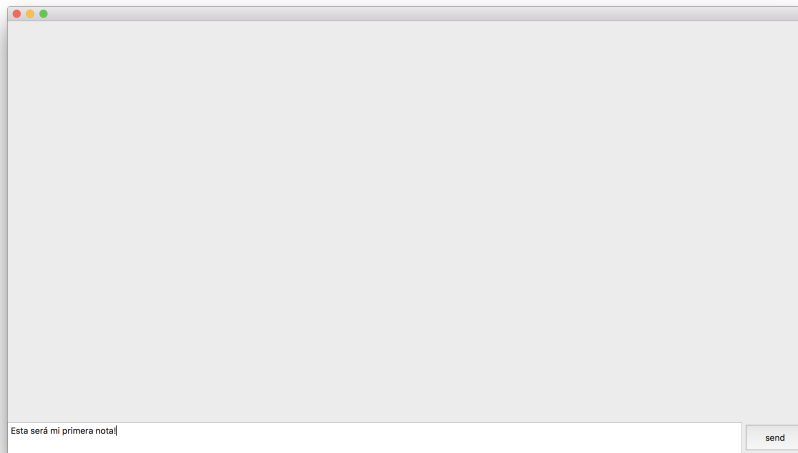
Introducción

Ya va quedando muy poco en el semestre y los ayudantes, estresados porque debemos dar vuelta todos nuestros ramos, decidimos ordenar las últimas correcciones del semestre de forma visual para que sus notas no se vean afectadas por nuestras otras tareas. Para esto, y como sabemos que ustedes son los mejores desarrolladores de PyQt5, les pedimos que creen una aplicación tipo *Post-it* en donde podamos agregar y eliminar notas.

Cómo usar la interfaz

Cómo agregar notas

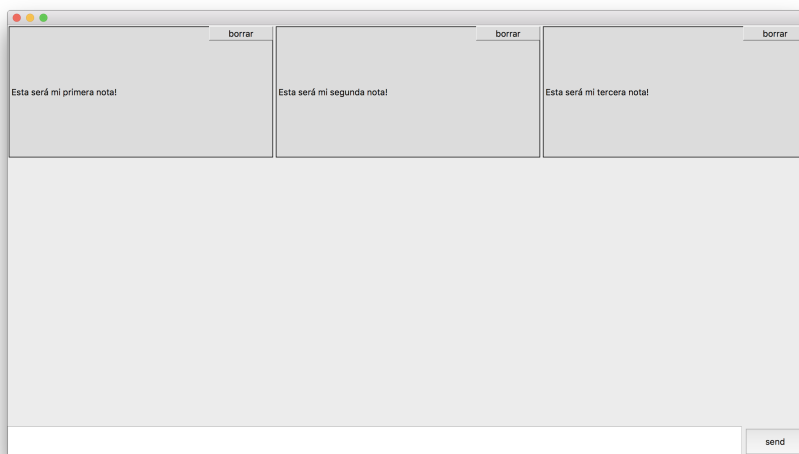
Se debe poder escribir texto en una sección, incluyendo saltos de línea.



Para que luego, a través de un botón, puedas *pegar* tu nota. Debería verse de la siguiente manera:

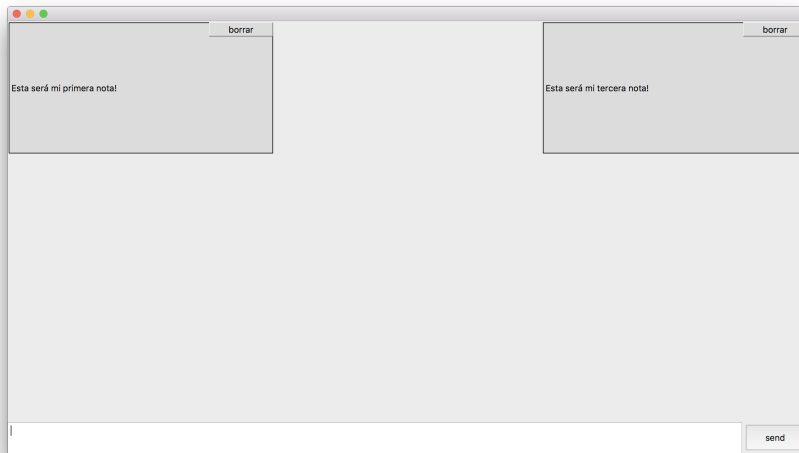


Nota que el texto recién ingresado debe borrarse de la sección en donde se edita luego de pegar tu nota. Luego de agregar un par de notas...

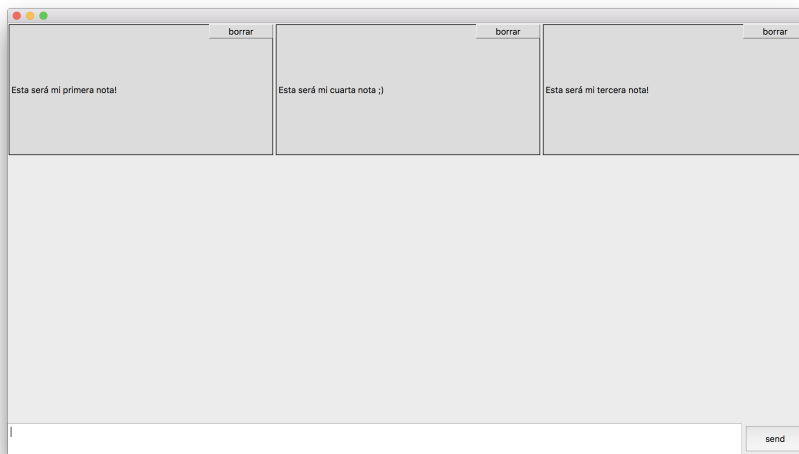


Cómo borrar notas

Cada nota debe tener un botón que al presionarlo borre la nota correspondiente, además se debe evidenciar visualmente qué hará cuando se presione ("X", *borrar*, *eliminar*, etc).



Luego de borrar la nota, debe quedar el espacio donde esta se encontraba libre de manera que, al ingresar una nota nueva, esta ocupe el primer lugar disponible.



Funcionamiento

- En esta actividad deberás construir un tablero para manejar tus notas/cosas por hacer/ideas. El tablero debe permitir agregar notas nuevas y eliminar notas existentes desde un botón que se encuentre **dentro** de la nota.
- Debe existir un lugar para escribir tus propias notas junto con un botón que al ser apretado, las envíe al tablero.
- Cuando se elimine una nota, su lugar debe quedar vacío.
- Al agregar nuevas notas estas deben ir al primer lugar disponible en el tablero.

Front-end

Estos son los componentes mínimos que deben verse en tu *front-end*:

- Un tablero que contenga hasta 9 notas al mismo tiempo, en forma de **grilla** de 3×3 .
- Un lugar para ingresar tus propias notas, junto con un botón propio.
- Cada nota debe tener un botón propio para eliminarla del tablero.

Back-end

Cada vez que se ingrese un ítem nuevo al tablero se deben revisar 4 cosas:

- La nota no puede estar vacía.
- El largo de la nota no debe superar los 140 caracteres.
- La nota no puede tener más de 3 saltos de línea para que se vea bien.
- No debe existir una nota exactamente igual en el tablero.

Front-end y back-end

Todo lo mencionado hasta ahora está referido al enfoque del usuario, es decir, a las funcionalidades del servicio. Sin embargo, es importante aprender a separar en *front-end* y *back-end*. Es obligatorio hacer la separación *front-end* y *back-end*. Como consejo, identifique cuales son las entidades que cambian su estado a través del programa y represéntelas en el *back-end*.

Recuerde que el *back-end* es la parte **lógica** del programa en la que se hacen los razonamientos y cálculos para cambiar los estados de las entidades, mientras que el *front-end* es sólo la representación visual del programa donde no se hacen grandes razonamientos. Se va a evaluar la correcta separación de ambos elementos.

Notas

- Puedes crear tu propia clase de *widget*, heredando de la clase `QWidget`, y así agregarle más funcionalidad.
- No se puede utilizar Designer.

Requerimientos

Para esta actividad **NO** está permitido usar Designer.

- (3.6 pts) Front-end
 - (0.8 pts) Se pueden escribir notas nuevas.
 - (0.8 pts) Al apretar el botón agregar, lo que se tiene escrito se va a alguna posición válida del tablero.
 - (0.6 pts) Si el tablero esta lleno no se permite agregar una nota nueva.
 - (0.7 pts) El botón para eliminar es parte de la nota que elimina.
 - (0.7 pts) Al apretar el botón los cambios se ven reflejados correctamente.
- (2.4 pts) Back-end
 - (0.6 pts) Se separa el *front-end* del *back-end*.
 - (0.2 pts) No se pueden crear notas vacías.
 - (0.4 pts) No se permiten notas con más de 140 caracteres.

- (0.4 pts) No se permiten notas con más de 3 saltos de líneas.
- (0.4 pts) No se permite crear una nota igual a una ya existente.
- (0.4 pts) Las notas se guardan y manejan a través de una estructura de datos ordenada.

Entrega

- **Lugar:** En su repositorio de GitHub en la **carpeta** `Actividades/AC10/`
- **Hora:** 16:40
- Si está trabajando en pareja y ambos miembros suben la carpeta `Actividades/AC10/`, se corregirá una entrega al azar. Si ambos desean tener el código en sus repositorios, la carpeta que no se debe corregir tiene que llamarse `Actividades/AC10-pareja/`.