

Universidad de Guadalajara
Centro Universitario de Ciencias Exactas e Ingenierías
(CUCEI)



ETAPA 3

Curso: Seminario de Solución de Problemas de Algoritmia

Clave del curso: I5885

NRC: 60033

Calendario: 2020A

Sección: D09

Profesor: Sencion Echauri Felipe

Alumno: Robles Perez Cristian Ismael

Código del alumno: 215472278

Guadalajara, Jalisco, México.

11 de marzo del 2020

Desarrollo:

El primer paso para la implementación de esta etapa es la creación de un `QScrollArea` desde el designer dentro de la segunda página del `QStackedWidget`, además se creará un widget temporal en el que se aplicará un layout que contiene todos los widgets con datos.

Para la correcta lectura de dichos datos primero se tiene que crear atributo de la clase de tipo `QJsonArray`, el cuál servirá de manera similar al atributo utilizado en la etapa anterior que guardaba un array con todos los datos de la sección “users” del archivo JSON con la diferencia de que ahora guardará la sección “products”.

Ahora se debe de crear la clase widget que almacenará la información leída, dicho widget contará únicamente con tres `QLabels`, uno para la imagen, otro para la descripción del producto y otro para el precio. Además, dicha clase contará con tres “setters” que tomarán la información correspondiente a cada campo y la modificarán para que dicha información se muestre correctamente, en el caso de la imagen se debe de concatenar el id del producto a la extensión “.jpg” además de añadir la ruta de la imagen y utilizar dicha información para la creación de un objeto `Pixmap`, en el caso del precio se concatena un “\$” al inicio y en el caso de la descripción simplemente se añade sin ningún tipo de cambio.

Una vez creada la clase anterior se necesita crear un método capaz de obtener cada uno de los datos del `QJsonArray` y crear su respectivo widget, para eso se itera por todo el `QJsonArray` de productos utilizando un `JsonObject` como auxiliar, además se debe de añadir dicho widget a un `QGridLayout` (el cuál es atributo de la clase). Al final del ciclo simplemente se utiliza el método `setLayout` del widget temporal del `QScrollArea` con el `QGridLayout` como parámetro. Dicho método se llamará “showProducts”

Finalmente, para la búsqueda por categoría se creó un `comboBox` en el designer y se le asignó un slot (“currentIndexChanged”) el cual mandará a llamar a la clase `showProducts` con las letras de cada id que corresponden a su categoría como parámetros o un asterisco en caso de que se quieran mostrar todos los departamentos ya que dentro de dicha función se compara el id de cada producto leído y si hay una coincidencia entre las primeras letras de éste y el parámetro proporcionado se añadirá dicho producto al layout.

Conclusión:

Definitivamente ésta fue una práctica mucho más retadora que las dos anteriores, ya que nunca antes había trabajado con layouts dinámicos y sus respectivos widgets ni con imágenes en archivos por lo que probé e investigué bastante para poder implementarlos de manera correcta. Sin embargo, considero que fue una práctica bastante interesante y definitivamente los temas implementados son de una gran utilidad.