# APÉNDICE C: Aplicación

El desarrollo de la aplicación trajo aparejado el uso de varias tecnologías, lenguaje de programación, interfaz gráfica, entorno de virtualización, aplicación de simulación, etc.

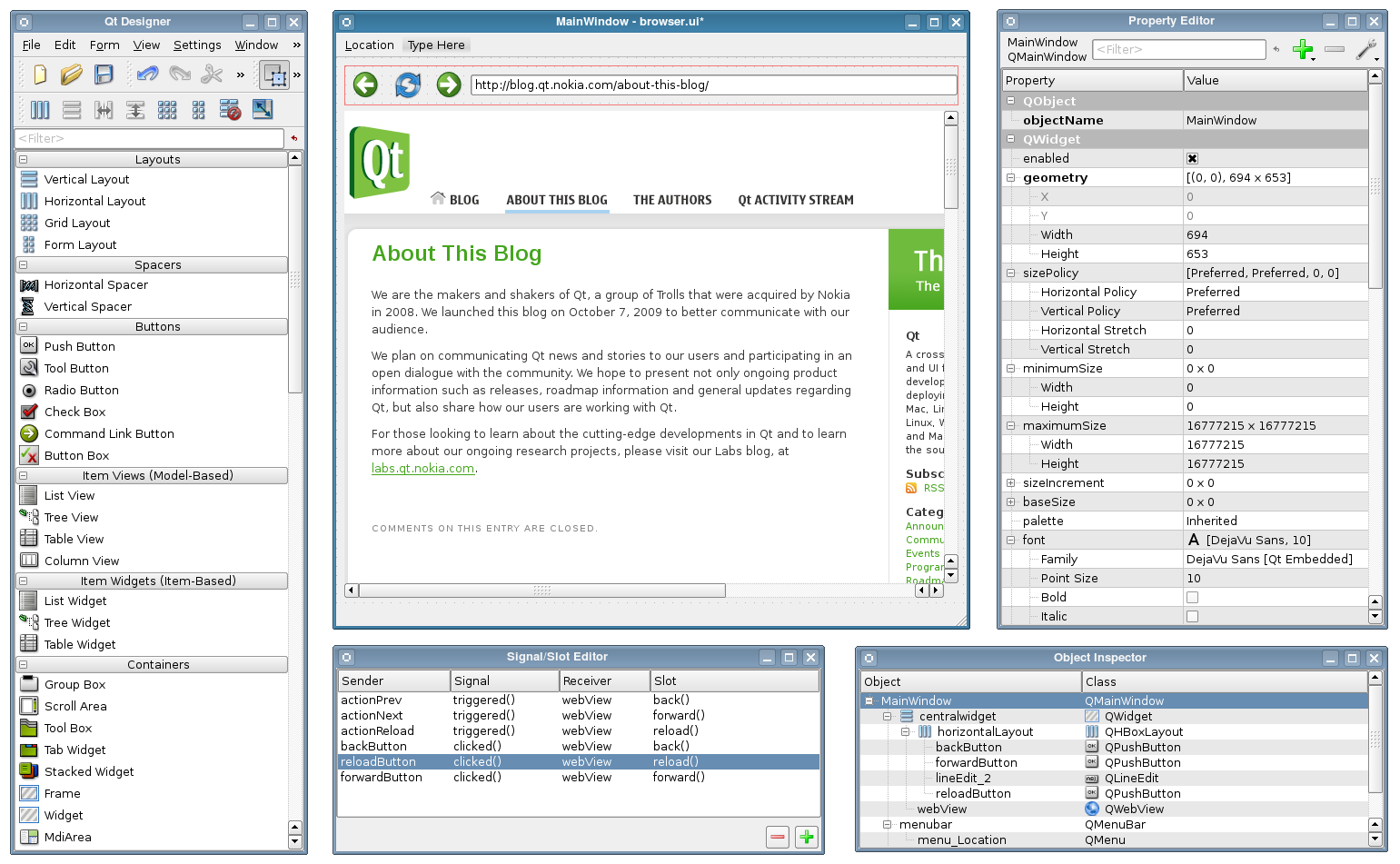
|  |  |
| --- | --- |
| Lenguaje de programación: | python-logo-master.png |

Comenzando por la codificación de la aplicación, el lenguaje de programación elegido fue Python. La elección de este lenguaje fue debido a las características presentadas en la siguiente lista.

|  |  |  |  |
| --- | --- | --- | --- |
| Interfaz gráfica: | Qt_logo002.png | y | 164px-Python_and_Qt_svg.png |

Debido a que la aplicación principal se encuentra escrita en Python, para la interfaz gráfica (GUI, Graphical User Interface) se utilizo un *“binding”* de la biblioteca grafica de Qt para Python llamado PyQt. Un “binding” es una adaptación de una biblioteca para ser usada en un lenguaje de programación distinto de aquél en el que ha sido escrita. En este caso utilizamos PyQt (binding) para acceder a los elementos de la biblioteca multiplataforma de Qt y desarrollar las interfaces gráficas de usuario ya que Qt se encuentra escrita en C++.

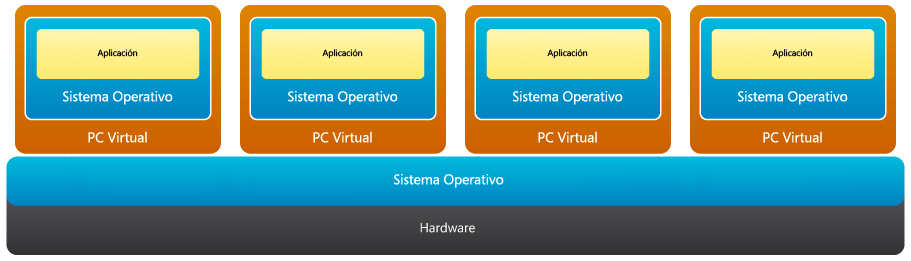
El entorno de desarrollo integrado (IDE por sus siglas en inglés) utilizado es Qt Designer. Esta IDE es muy intuitiva y de fácil uso permitiendo un diseño rápido y sencillo. En la figura siguiente se observa en la barra lateral izquierda los elementos de la biblioteca grafica para agregar. En el panel central se encuentra la *“Main Window”* o ventana principal, donde se arma la interfaz grafica propiamente dicha. En la barra lateral derecha se encuentra el editor de propiedades, este nos permite definir diversas opciones de cada uno de los elementos agregados en la ventana principal como ser titulo, tamaño, etc. En la parte inferior se observan dos ventanas mas, la izquierda es la ventana de edición de señales y ranuras, con ella se asignan ranuras (acciones que la aplicación realiza al recibir la señal) para cada una de las señales que la GUI recibe; la derecha es la ventana de inspección de objetos, brinda una lista en forma de árbol con las relaciones de todos los elementos de la biblioteca utilizados según su ubicación dentro de la ventana principal.



|  |  |  |  |
| --- | --- | --- | --- |
| Ambiente de simulación: | vmware_logo.png | y | pspice_logo.gif |

El ambiente de simulación fue un elemento crítico a la hora de realizar la inyección automatizada. Los recursos computacionales eran limitados y las simulaciones a realizar eran más de 56.000. Se utilizaron dos computadoras físicas para realizar las simulaciones automáticas. Una de ellas con un procesador Intel® Core™2 Quad(4 núcleos) funcionando a 3.00 GHz, y la otra con un procesador Intel® Core™2 Duo(2 núcleos) a 2.40 GHz, ambas con tecnologías Intel® Virtualization Technology (Intel® VT) y Intel® Virtualization Technology (Intel® VT) for Directed I/O (Intel® VT-d). Esto permitió un rendimiento superior y reducciones de hasta 5 veces en los tiempos de simulación.

En la siguiente imagen se puede apreciar gráficamente el concepto de ambiente virtualizado.



En este caso, la imagen muestra la implementación sobre el procesador de 4 núcleos. La decisión de utilizar virtualización fue debida a que la aplicación de simulación, OrCAD PSpice A/D, no hace uso de todos los procesadores disponibles. De esta manera, al crear 4 computadoras virtuales utilizando la tecnología Vmware se logró ejecutar 6 instancias paralelas de la aplicación (1 computadora por núcleo físico).

|  |  |
| --- | --- |
| Almacenamiento y versionamiento: | Google_Code_logo.png |

Para el desarrollo de la aplicación y almacenamiento de datos se utilizo una herramienta de versionamiento llamado Subversion, y como almacenamiento del código de la aplicación Google® Code. Esto permitió un desarrollo progresivo de la aplicación y también la sincronización en el almacenamiento de los resultados de cada una de las 6 computadoras virtuales. Actualmente el proyecto se encuentra alojado bajo la licencia GNU General Public License v3. Los hipervínculos son los siguientes:

* Página principal del proyecto: <http://code.google.com/p/fiocs/>
* Descarga directa de la última versión: [fail\_injector-0.5.6.exe](http://fiocs.googlecode.com/files/fail_injector-0.5.6.exe)
* Licencia: <http://www.gnu.org/licenses/gpl.html>