Capitulo 5. Diseño de MACLEN

A continuación se indica la tecnología JavaFX que se requirió para el desarrollo de la aplicación MACLEN (Modalidad de Aprendizaje en Celulares para Lenguajes), el ambiente de desarrollo elegido es Netbeans 6.7 que trabajo en conjunto con el administrador de servicios XAMPP.

5.1 Ambiente de desarrollo

NetBeans 6.7.1 [6] es el Ambiente de Desarrollo Integrado (integrated development environment, *IDE*), con el cual se puede realizar el desarrollo de software sobre diferentes lenguajes de programación tales como: C/C++, Java, JavaFX, PHP, JavaScript, Groovy y Ruby. Soporta los Servidores: GlassFish, TomCat entre otros y Base de Datos en MySQL, es de libre distribución y *open source*.

NetBeans es el IDE recomendado por la plataforma JavaFX para realizar el desarrollo de aplicaciones, por contar con un diseño de soporte de la tecnología JavaFX. Permite realizar la instalación de un conjunto de plugins necesarios que complementan las funcionalidades y servicios que pretenda proporcionar la aplicación a desarrollarse.

5.2 Servicio Web y JavaFX

JavaFX interactúa con Java al requerir de un Servlet y un Servicio Web, por medio del envío y recepción de peticiones en formato XML. El Servicio Web es el encargado de realizar la consulta a la Base de Datos en MySQL y enviar la respuesta en formato XML, realizándose la interacción por medio de un Servlet. NetBeans proporciona las condiciones necesarias para que la interoperabilidad se realice sin pérdida de información o caída de servicios, el IDE está diseñado para soportar estas herramientas de desarrollo.

5.3 MySQL

MySQL es un sistema de gestión de Base de datos relacional, multihilo y multiusuario, actualmente forma parte de Sun Microsystems, es de libre distribución. En NetBeans se encuentra como una de las Base de Datos de preferencia, contando con documentación para su configuración y ejecución.

El Servicio Web presente en MACLEN, realiza las consultas a la Base de Datos en MySQL, siendo rápida y sin pérdida de datos o conexión.

5.4 GlassFish

Es un servidor de aplicaciones, forma parte de Sun Microsystems, es de código libre y gratuito, el IDE NetBeans instala y configura automáticamente GlassFish. El Servicio Web utilizado en MACLEN es ejecutado sobre el servidor GlassFish, utiliza la dirección *localhost.*

5.5 Diseño

JavaFX es un conjunto de tecnologías de Sun, destinadas a la creación de Aplicaciones de Internet Enriquecidas (*Rich Internet Applications,* RIA). Las RIA son todas las aplicaciones Web de última generación que contienen una interfaz compleja y funciones propias de las aplicaciones de escritorio, que pueden ser ejecutadas en dispositivos tales como televisión, navegador de Internet y teléfono celular.

La forma de realizar las declaraciones en JavaFX, como se indicó en el capítulo 4 difiere de Java, ya que la forma de trabajar con las clases, funciones e instanciaciones es diferente, siendo el elemento básico el nodo, con el cual se representa un elemento grafico en pantalla.

5.6 Arquitectura general de MACLEN

MACLEN esta divido en cuatro paquetes: Paquete MACLEN, paquete model, paquete view y paquete parser. El diseño de la arquitectura presente está basado en el tutorial Media Browser de JavaFX [23].

- Paquete *maclen*: Esta conformado por todas las clases que ayudan a crear los tres objetos de aprendizaje móvil, los cuales son de tipo: Vocabulario, Patrón gramatical y Expresiones.

- Paquete *model*: Contiene el modelo de representación de la información obtenida a través de `HTTP GET`.

- Paquete *view*: Controla los diferentes estados y funciones que presenta un botón.

- Paquete *parser*: Encargado de manipular los datos obtenidos a través del Servicio Web y el protocolo `HTTP GET`.

En la Figura 5, se puede observar la descripción de la arquitectura presente en MACLEN

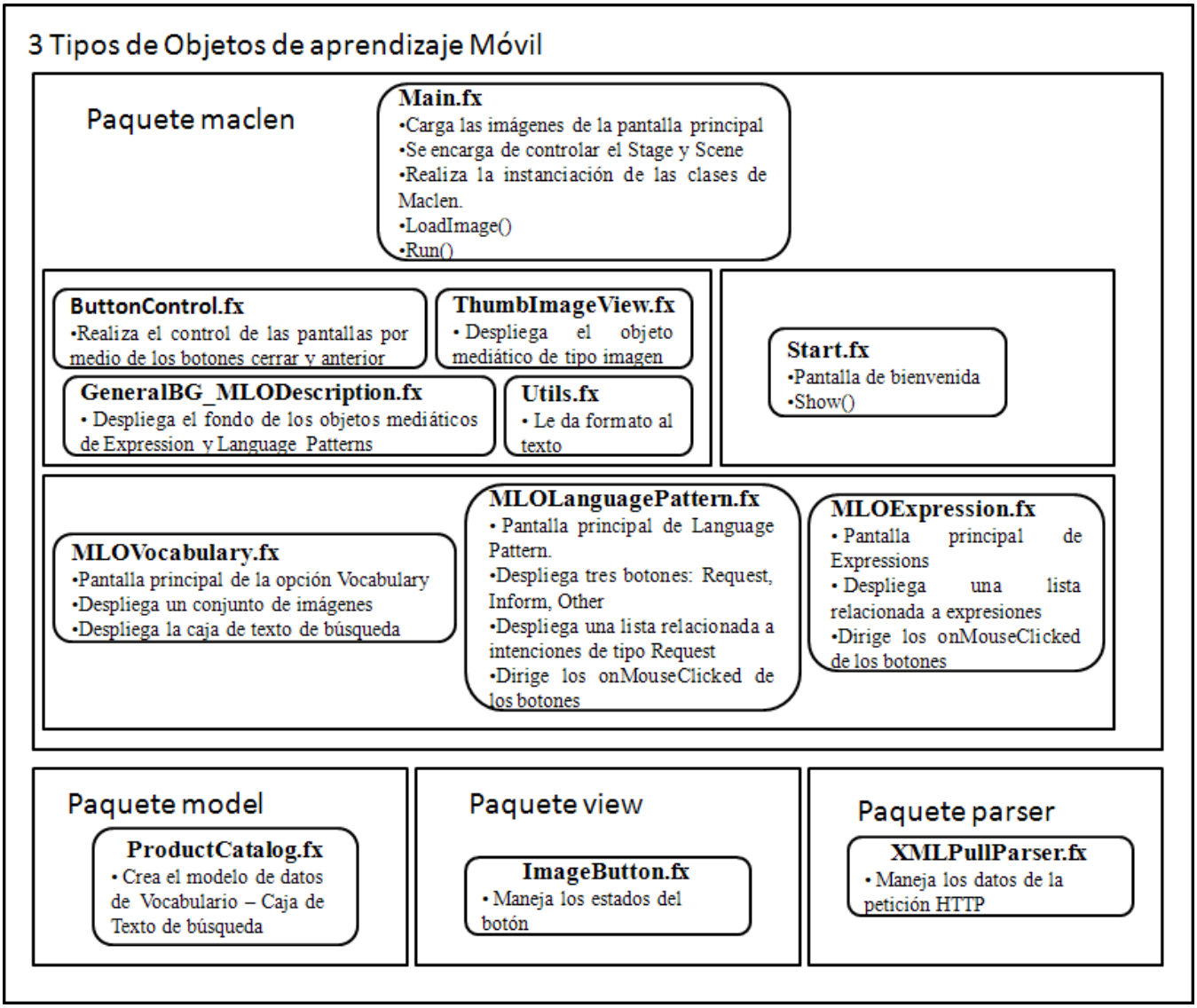

Figura 5. Arquitectura de MACLEN.

Arquitectura del módulo *Vocabulary*

La arquitectura correspondiente al módulo de *Vocabulary* contiene un conjunto de clases, las cuales se encargan de realizar la manipulación del objeto de aprendizaje móvil de tipo Vocabulario. El módulo consiste en dos partes principales que son: *Categories* y *Search for*, como se puede observar en la Figura 6.

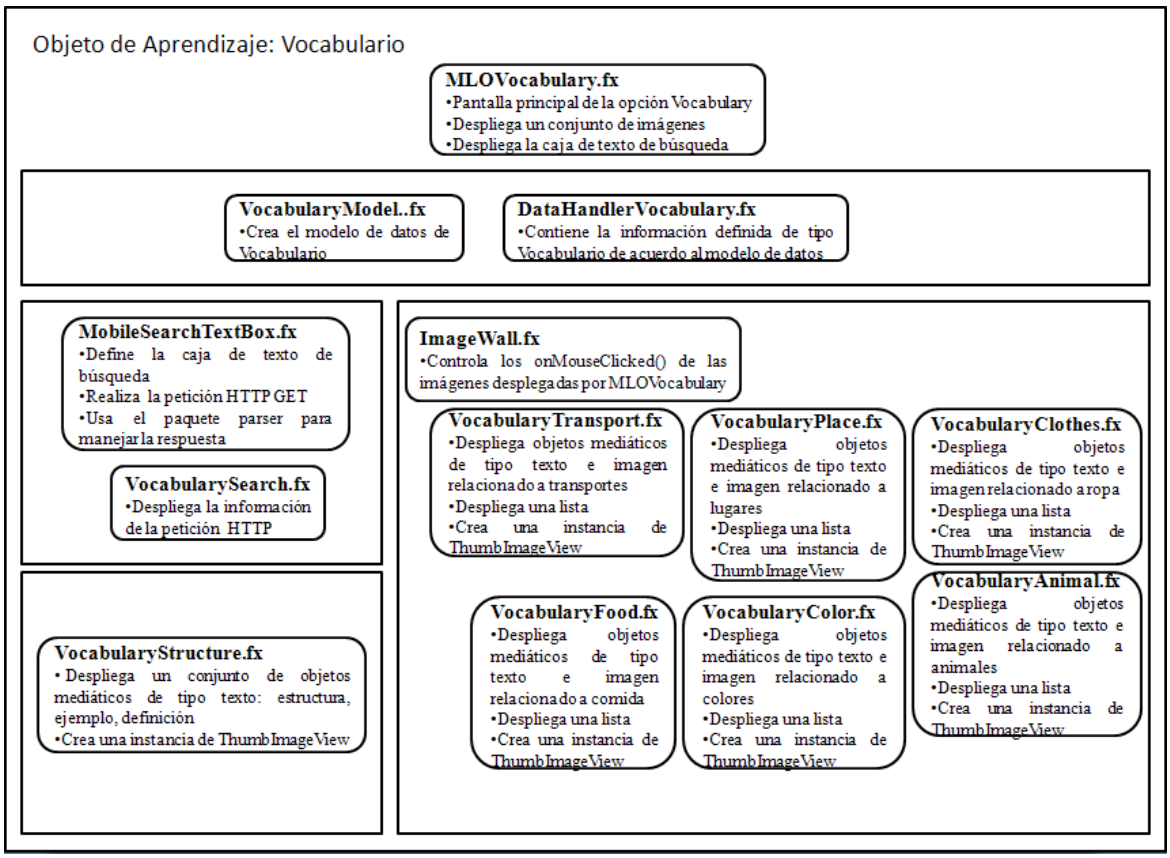

Figura 6. Arquitectura de *Vocabulary*.

Arquitectura del módulo *language patterns*

El módulo de *language patterns* está conformado por tres tipos de intenciones: *Request*, *Inform*, *Other*, la arquitectura que la representa se puede observar en la Figura 7.

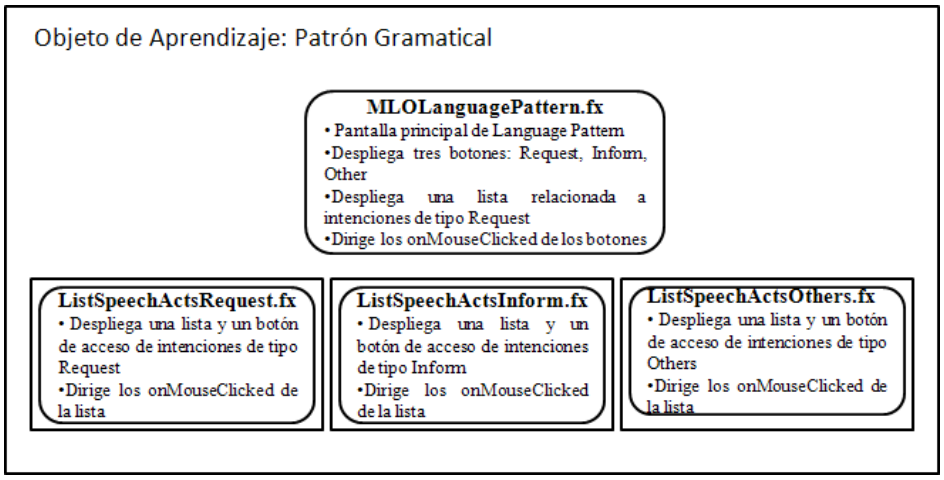

Figura 7. Arquitectura principal del módulo *language patterns*.

Arquitectura de *request*

A continuación se indica en la Figura 8, la arquitectura encargada de realizar la representación de la intención de tipo *Request*.

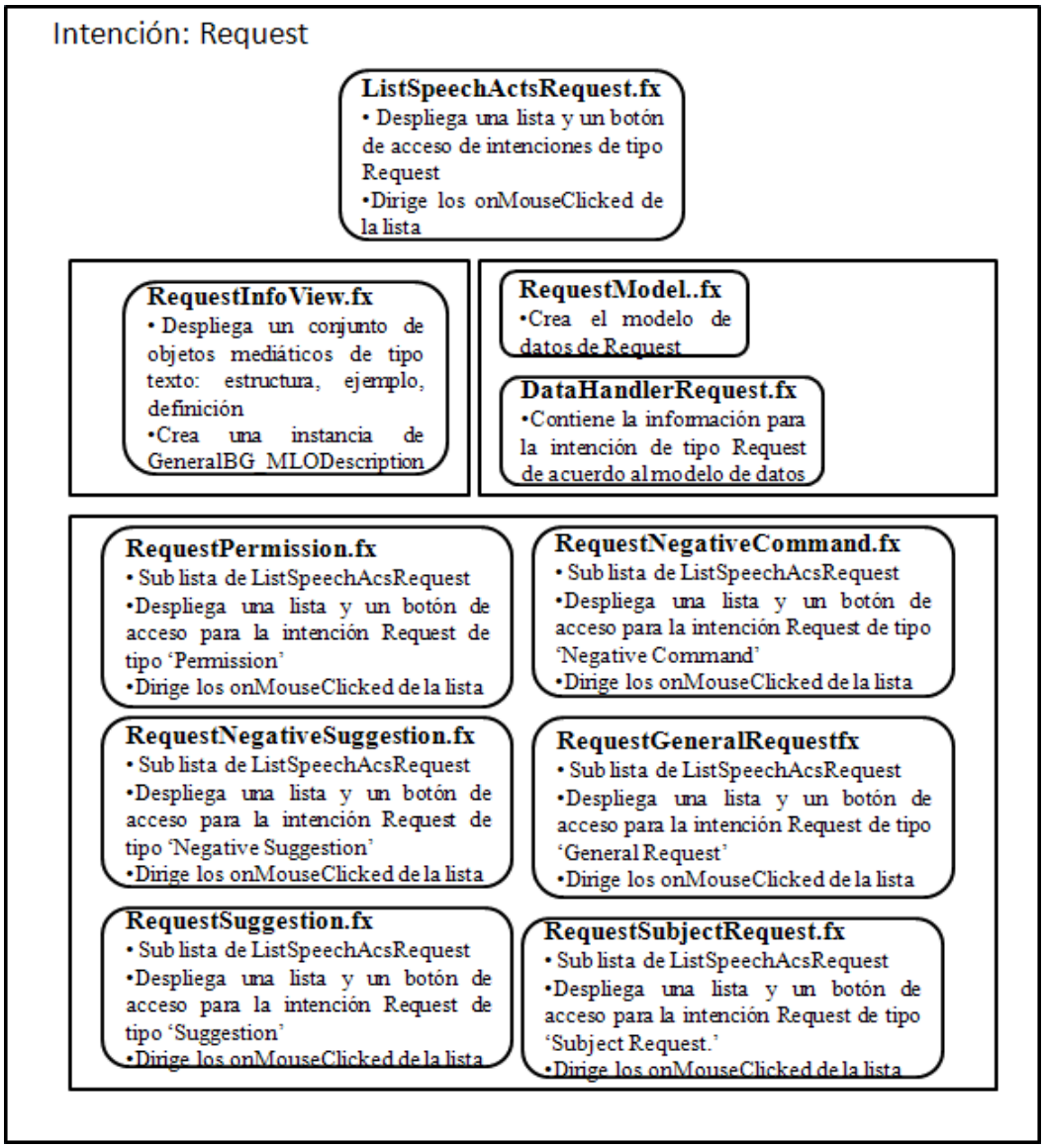

Figura 8. Arquitectura de la intención *Request.*

Arquitectura de *Inform*

A continuación se indica en la Figura 9, la arquitectura encargada de realizar la representación de la intención de tipo *Inform*.

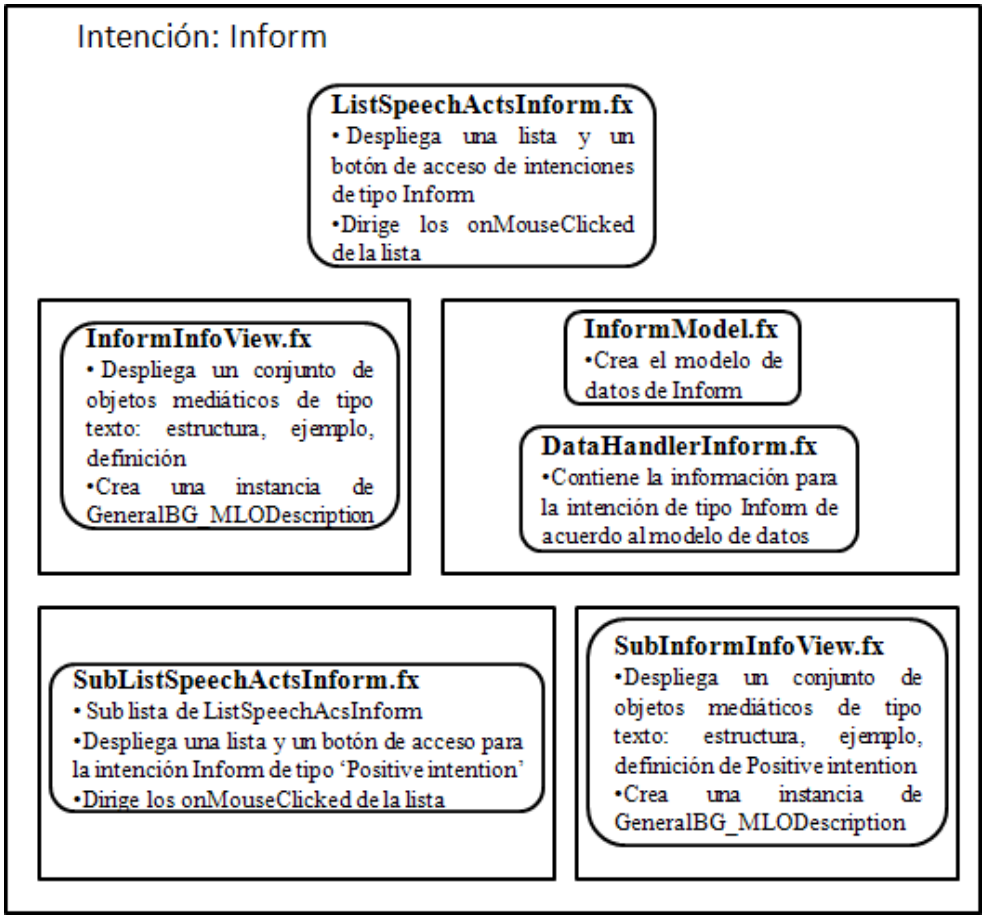

Figura 9. Arquitectura de la intención *Inform.*

Arquitectura de *Other*

A continuación se indica en la Figura 10, la arquitectura encargada de realizar la representación de la intención de tipo *Other*.

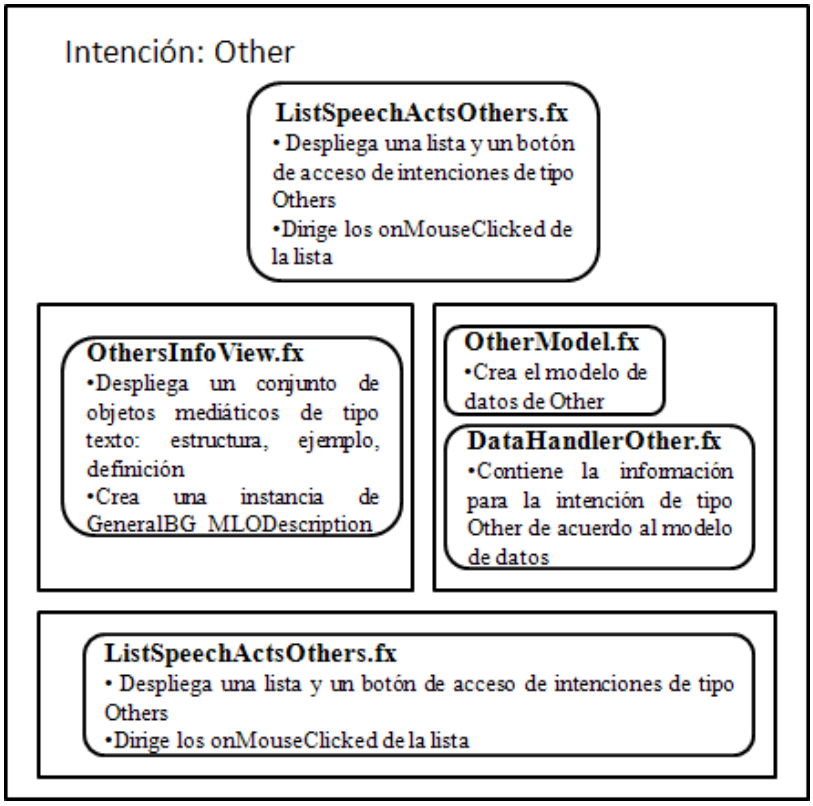

Figura 10. Arquitectura de la intención *Other*.

5.7 Arquitectura del módulo *Expressions*

El conjunto de clases presentes en la Figura 11, conforman la arquitectura del módulo de *Expressions*.

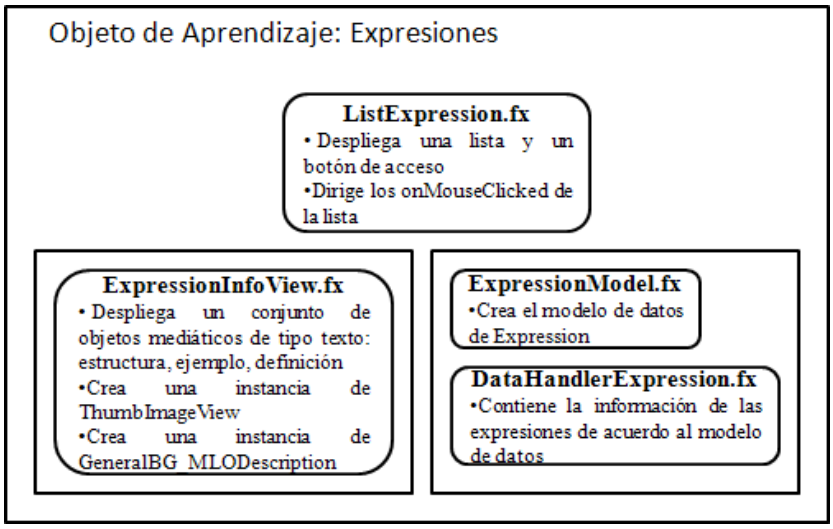

Figura 11. Arquitectura del módulo *Expressions.*

5.8 Diseño de aplicaciones en JavaFX

Para el diseño de la arquitectura a utilizar en el desarrollo de una aplicación se recomienda tomar en cuenta lo siguiente:

- **`Paquete`.-** El paquete está conformado por un conjunto de `CustomNode` o clases que realizan funciones en común, por lo tanto se deben de determinar las funciones generales que va a llevar a cabo la aplicación, para determinar el o los paquetes a

crearse, asignando a cada paquete un nombre descriptivo y corto para su fácil localización.

- **`CustomNode`.-** Denominado también clase, utilizado para la definición del modelo de datos, definición de objetos a visualizarse en pantalla, un `CustomNode` puede instanciar un conjunto de `CustomNodes` para crear uno más complejo y completo.

- **Main.fx**.-Clase principal, es la primera que ha de ser construida, por encargarse de realizar la instanciación de las clases que conforman la aplicación, controla a `Stage` y a `Scene`, del Main depende el objeto a visualizarse en pantalla.

Estos elementos son los que se toman en cuenta al momento de realizar el diseño de la arquitectura de la aplicación MACLEN.