

7 Conclusiones

La realización del presente proyecto me ha permitido aprender a modelar sistemas software usando el método UML, y la familiarización con una aplicación muy usada para la realización de tales modelos como es Rational Rose.

Por otra parte ha sido una muy buena la experiencia de llevar a la práctica los conocimientos teóricos de C++ que se imparten en la carrera. Cuando hay que ponerse a generar código, es cuando se es consciente de las lagunas que se tienen y cuando realmente se aprende a programar.

También me ha permitido familiarizarme con uno de los entornos de desarrollo visuales más extendidos. Esta experiencia ha supuesto una grata sorpresa al descubrir cuánto simplifica este entorno el desarrollo de interfaces gráficas. Incluye una librería de clase visuales (VCL) realmente completa que permite crear fácilmente todos los objetos gráficos que se pueden necesitar para crear el interfaz de cualquier aplicación.

Aunque sin duda lo mejor es este entorno, al menos en mi opinión, es la abstracción que proporciona sobre la gestión de eventos en entornos Windows: en cualquier programa desarrollado para Windows, el programador debe capturar los distintos mensajes para actuar en consecuencia. Sin embargo, a los objetos creados usando la VCL de Borland se le pueden asociar funciones que deben ejecutar ante la llegada de ciertos eventos, librando al programador de la tarea de procesar todos los mensajes del sistema.

Finalmente, destacar que ha sido algo decepcionante comprobar la incapacidad del software Rational Rose para generar modelos a partir del código fuente desarrollado con C++ Builder. El único entorno de desarrollo soportado por Rational Rose era Microsoft Visual C++.