



# TESINA DE LICENCIATURA

**Título:** Lenguajes formales y derivación automática de código de pruebas a partir de modelos de software con restricciones OCL

**Autores:** Ilan Rosenfeld

**Director:** Claudia Pons

**Codirector:** -

**Asesor profesional:** Gabriela Alejandra Perez

**Carrera:** Licenciatura en Informática

## Resumen

Qué testear es un tema siempre vigente. Con tal propósito, y en caso de querer derivar casos de prueba automáticamente desde un modelo de software, no contamos con la precisión necesaria en los mismos para generar los tests acordes a nuestros intereses. Lenguajes formales como OCL permiten enriquecer cualquier modelo mediante información adicional o restricciones sobre sus elementos; entonces, la derivación de código y casos de prueba a partir de un modelo enriquecido con restricciones y especificaciones en este lenguaje permitirá contar con un soporte mucho más robusto de nuestro sistema.

Tras el análisis de varias herramientas de generación automática de código a partir de modelos de software, se llegó a la conclusión de que casi ninguna de ellas incluye la traducción de restricciones en el modelo escritas en un lenguaje formal. Es por ello que la tesina consistió en un análisis de varios lenguajes formales de especificación (o modelado), especialmente de los lenguajes UML/OCL y Alloy. A partir de ello se desarrolló una herramienta para Eclipse, que permite, a partir de una especificación de un modelo UML poseyendo restricciones OCL, la generación automática de código Java, incluyendo las clases del modelo junto con sus respectivos Casos de Prueba, regulados por OCL. Paralelamente se genera de forma automática una especificación Alloy que permite el análisis formal estático del modelo.

## Palabras Claves

Acceleo, Alloy, AlloyMDA, Desarrollo de Software dirigido por Modelos (MDD), EasyMock, Eclipse, Java, JUnit, Mockito, OCL, Papyrus, Pruebas de Software Dirigidas por Modelos (MDT), PSM, Traducción de código, U2TP, UML, Verificación formal.

## Trabajos Realizados

Investigación teórica del MDD, del MDT y de lenguajes de especificación formal, específicamente OCL y Alloy. Desarrollo de código de traducción automática Acceleo para generar casos de prueba Java a partir de un modelo de datos con restricciones OCL. Uso y aprovechamiento de la herramienta AlloyMDA para verificación formal del código resultante.

## Conclusiones

Se creó una herramienta que permite traducir un modelo de datos con restricciones formales a su correspondiente código Java, automatizando la generación de casos de prueba robustos y especificados no sólo en este lenguaje sino también en dos lenguajes formales (OCL y Alloy), lo que le da un soporte confiable y verificable con diversas técnicas. Se lograron integrar distintos lenguajes, cada uno con su sintaxis propia, para concluir en un resultado que es más que utilizable por el usuario que así lo desee y que le permite detectar inconsistencias en su modelo.

## Trabajos Futuros

Permitir especificar las restricciones en el modelo fuente directamente en el lenguaje Alloy; lograr que Acceleo regenere el código sin alterar las actualizaciones realizadas o aquel texto que estuviera delimitado por separadores especiales; contar con tests menos abstractos y desprenderse de a poco del uso de mocks, para generar tests más específicos; generar contraejemplos en lenguaje natural/Java y soluciones posibles a las inconsistencias; realizar el mismo proceso permitiendo la selección de diversos lenguajes de programación destino.