



Universidad Nacional
Abierta y a Distancia

MODELADO DE DATOS

Modelando con StarUML

Descripción

Se realiza una breve explicación sobre el programa StarUML, sus herramientas e instalación.
Se diagrama la estructura lógica del ejercicio propuesto en la Lectura 4.



Luis Manuel Márquez Rodríguez
Ingeniería de Sistemas



Tabla de Contenido

	pág.
StarUML	2
Descarga e Instalación	3
Ejercicio	6
Referencias Bibliográficas	11

StarUML

Es una herramienta para el modelado de software basado en el Lenguaje Unificado de Modelado (UML) y la Arquitectura Dirigida por Modelos (MDA, por sus siglas en inglés Model Driven Architecture), inicialmente era un producto comercial conocido como Plastic, pasando en el año 2005 a ser de licencia abierta GNU/GPL.

Entre sus características se encuentran:

- Soporte completo al diseño UML mediante el uso de:
 - Diagrama de casos de uso
 - Diagrama de clase
 - Diagrama de secuencia
 - Diagrama de colaboración.
 - Diagrama de estados
 - Diagrama de actividad.
 - Diagrama de componentes
 - Diagrama de despliegue.
 - Diagrama de composición estructural (UML 2.0)
- Definir elementos propios para los diagramas, que no necesariamente pertenezcan al estándar de UML,
- La capacidad de generar código a partir de los diagramas y viceversa, actualmente funcionando para los lenguajes c++, c# y java.
- Generar documentación en formatos Word, Excel y PowerPoint sobre los diagramas.
- Patrones GoF (Gang of Four), EJB (Enterprise JavaBeans) y personalizados.

- Plantillas de proyectos.
- Posibilidad de crear plugins para el programa. (Black-Byte, 2006)

Descarga e Instalación

Procederemos a la descarga del programa desde su página oficial: <http://staruml.io/>

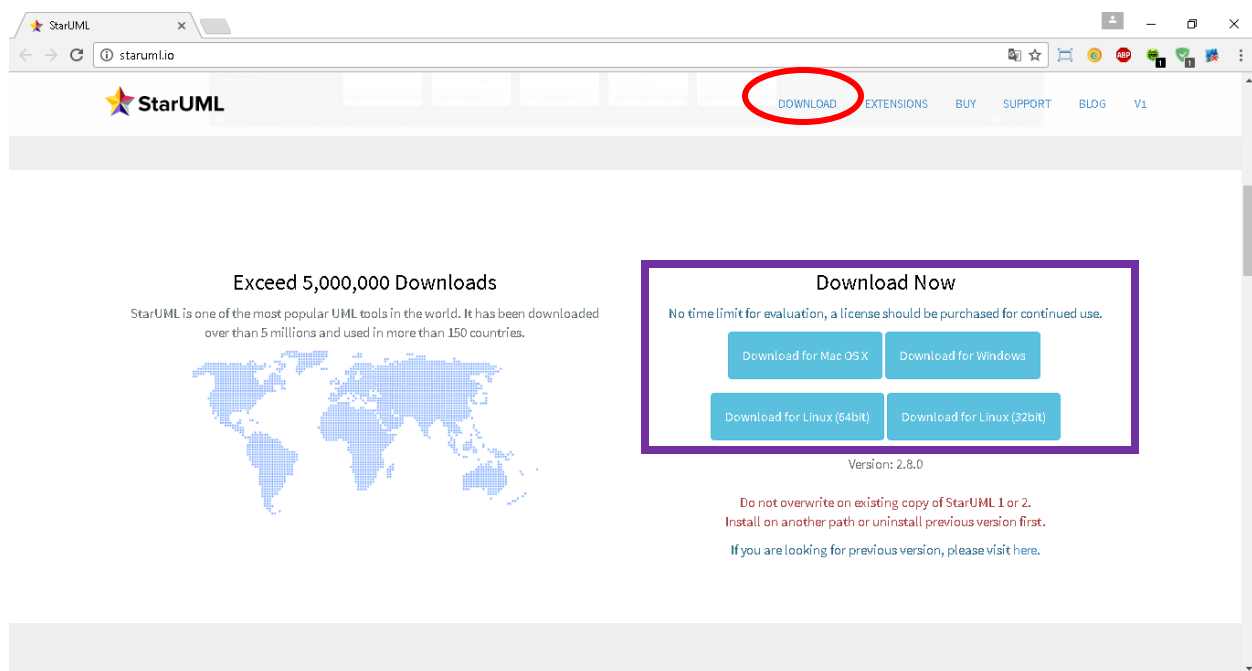
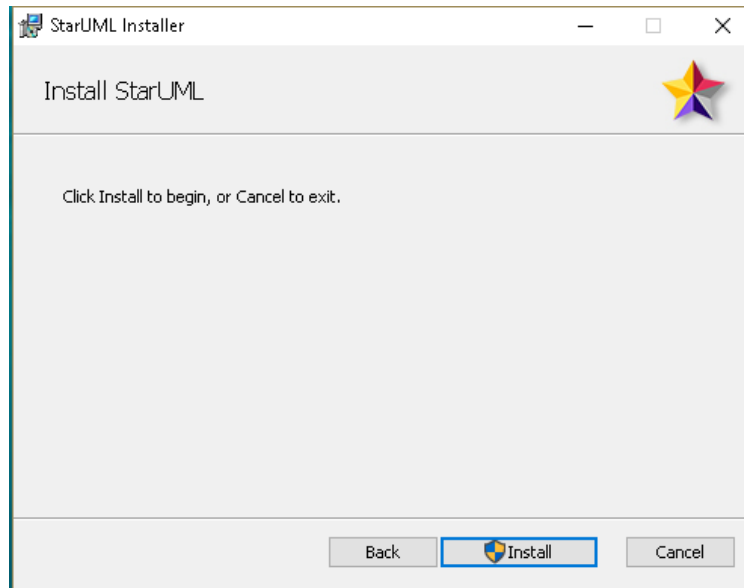
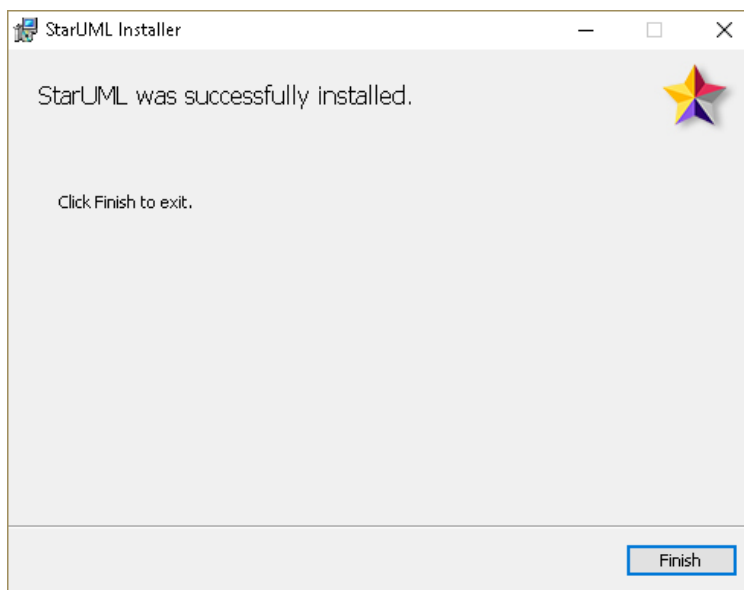


Figura 1. Página oficial de descarga del programa StarUML

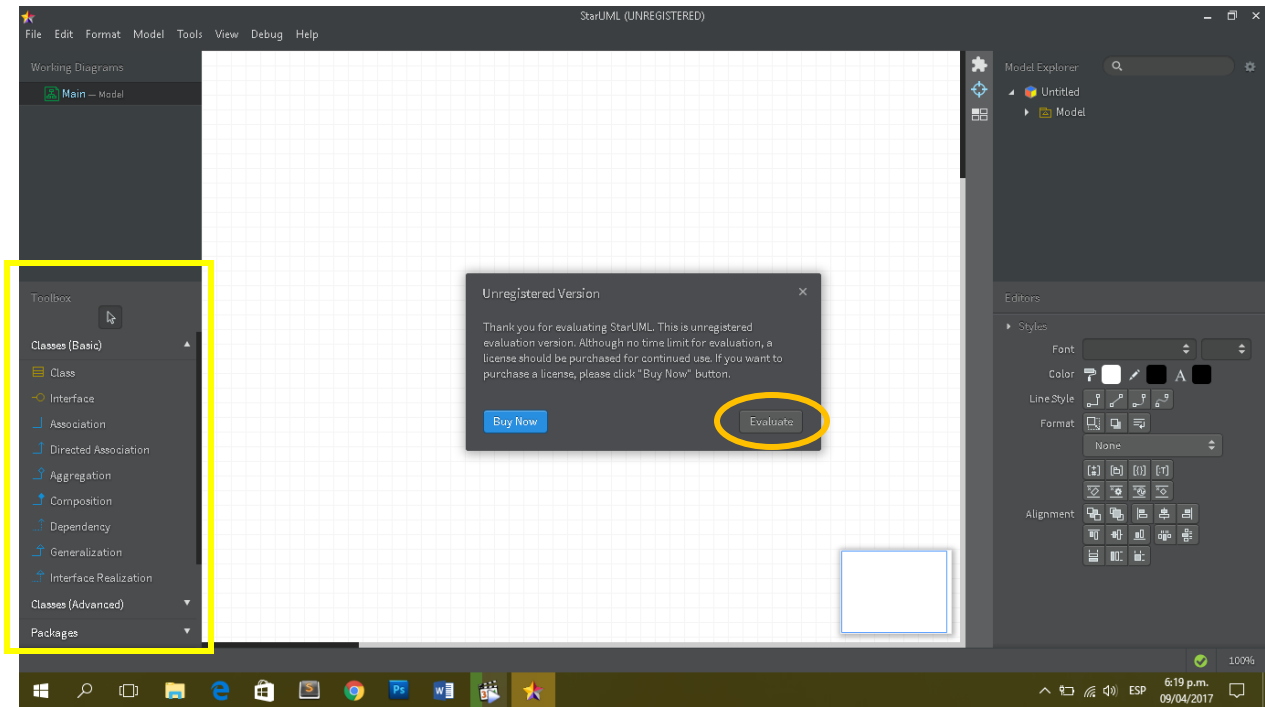
Una vez en la página, damos clic en la opción "DOWNLOAD", o bien buscamos la opción de descarga de acuerdo a nuestro sistema operativo.



Damos clic en Install y esperamos a que se complete la instalación.



Y clic en Finish para terminar. Luego abrimos el programa desde la barra de inicio. Aparecerá en el inicio una pequeña caja de diálogo, preguntándonos si deseamos la versión de prueba o la de pago, daremos clic en el botón 'Evaluate'.



A la izquierda se observa la barra de herramientas para los diagramas que requerimos para el desarrollo del ejercicio.

Ejercicio

Proponga un diagrama E-R para una base de datos para administrar la información relativa a algunos aspectos del campeonato mundial de fútbol considerando los siguientes supuestos:

- ✓ Un jugador pertenece a un único equipo y se identifica por: Código, Nombre, Alias, Nacionalidad.
- ✓ Es obligatorio en todo momento que un jugador pertenezca a un equipo determinado y no podrá cambiar de equipo a lo largo del mundial.
- ✓ Un equipo se identifica por: Código, Nombre, País.
- ✓ Cada partido involucra a dos equipos. De cada partido se necesita saber el resultado final, lugar de celebración, fecha, hora, árbitros que lo arbitran.
- ✓ En cada partido intervienen tres árbitros: 2 jueces de línea y un árbitro.
- ✓ Un árbitro puede realizar una función en un partido y otra distinta en otro partido.
- ✓ De cada árbitro se necesita conocer: código, nombre, nacionalidad

En primer lugar, identificamos las entidades y sus atributos.

Entidad	Jugador	Árbitro	Equipo	Partido
Atributos	Código	Código	Código	Código
	Nombre	Nombre	Nombre	Lugar
	Alias	Nacionalidad	País	Fecha
	Nacionalidad			Hora

Posteriormente damos nombre a cada atributo de cada entidad, y establecemos relaciones y su cardinalidad, generando así el diagrama ER.

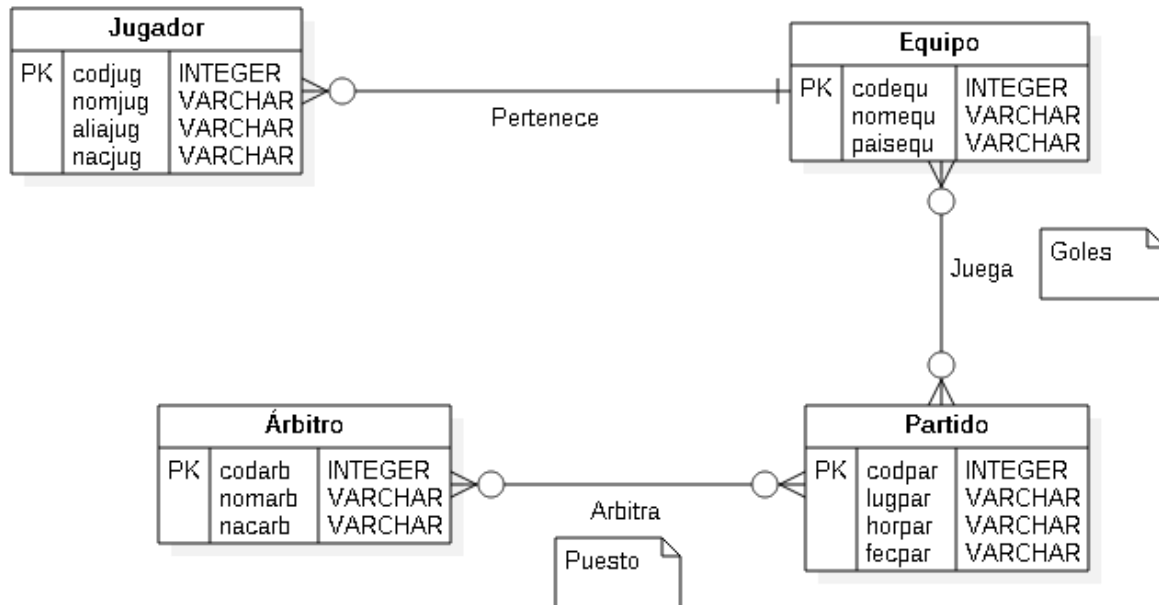


Figura 2. Diagrama entidad relación (ER)

Explicación

Las entidades pasan a formar tablas directamente:

- Partido (CodPartido, LugPartido, HorPartido, FecPartido)
- Equipo (CodEquipo, NomEquipo, PaiEquipo)
- Jugador (CodJugador, NomJugador, AliJugador, NacJugador)
- Árbitro (CodArbitro, NomAebitro, NacArbitro)

Las relaciones tienen el siguiente tratamiento:

a) Relaciones de grado mayor de dos:

La relación pasa a formar una tabla que contiene los identificadores de cada entidad involucrada y los atributos propios de la relación (si los hay).

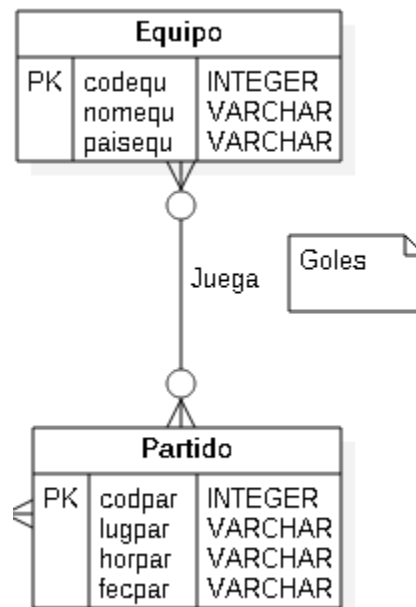
Para asociaciones grado dos:

b1) Relaciones muchos a muchos

En este tipo de relación también se origina una tabla que contiene los identificadores de cada entidad involucrada y los atributos propios de la relación (si los hay).



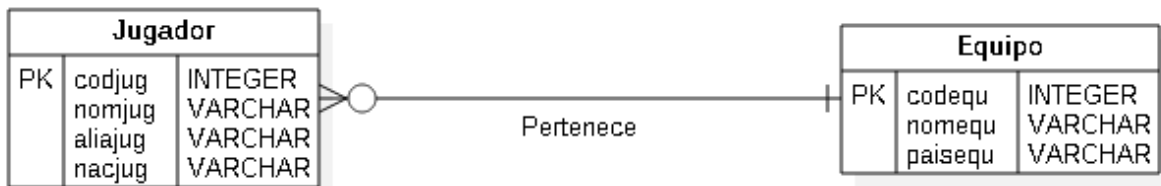
- Partido (CodPartido, LugPartido, HorPartido, FecPartido)
- Árbitro (CodArbitro, NomAebitro, NacArbitro)
- Partido_Arbitro (CodPartido, CodArbitro, Puesto)



- Partido (CodPartido, LugPartido, HorPartido, FecPartido)
- Equipo (CodEquipo, NomEquipo, PaiEquipo)
- Partido_Equipo (CodPartido, CodEquipo, Goles)

b2) Relaciones uno a muchos

En este tipo de relación se modifica la tabla de los hijos ya que heredan la llave del padre.



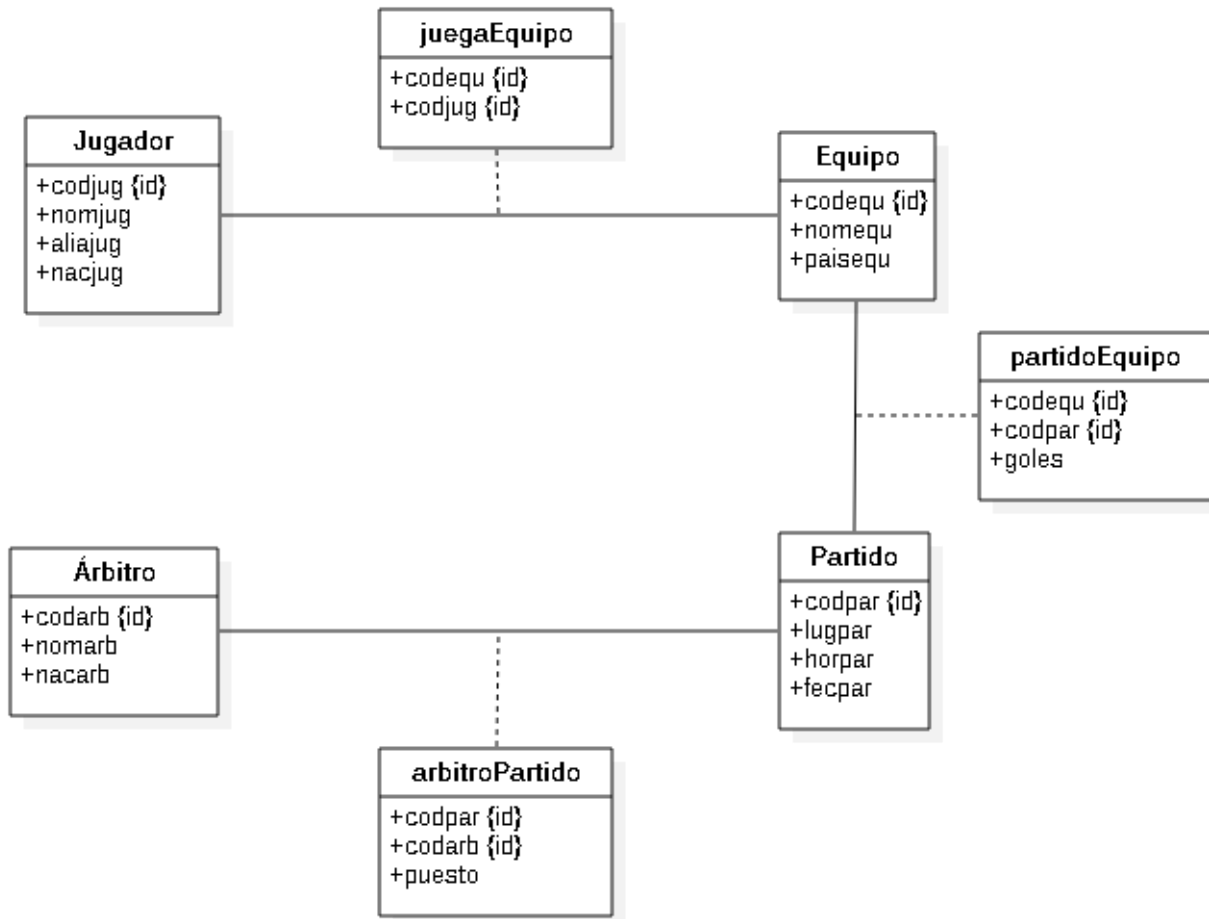
- Equipo (CodEquipo, NomEquipo, PaiEquipo)
- Jugador (CodEquipo, CodJugador, NomJugador, AliJugador, NacJugador)

b3) Relaciones uno a uno

A cualquiera de los dos se le agrega el identificador del otro (Normalmente el de menor cardinalidad). En este caso no se presentan.

Para realizar la transformación de un diagrama ER en un esquema relacional debemos:

- Transformar en tablas todos los tipos de entidad y relaciones que aparecen en el diagrama.
- Seleccionar las claves primarias para cada una de las tablas del esquema lógico.
- Fusionar las tablas que compartan su clave primaria.
- Normalizar el esquema resultante.
- Definir las restricciones de integridad aplicables al esquema obtenido.



Se han quitado las relaciones muchos a muchos.

No hay relaciones 1 a 1.

Referencias Bibliográficas

Black-Byte (19 de agosto de 2006). StarUML. Recuperado de <http://black-byte.com/review/staruml/>