

Proyecto II: Caso de Uso – Universidad

El objetivo de este proyecto es desarrollar un caso de uso mediano que para que utilicen conceptos del modelo relacional-objeto en Oracle.

Descripción del Problema

La Universidad Tecnológica (UT) mantiene una base de datos extensa con el propósito de soportar sus operaciones diarias. Esta base de datos incluye información pertinente a los campus, facultades, edificios, personal, grados y materias ofrecidas, y otros datos derivados de estos. La oficina de Servicios de Tecnología de Información (STI), que es responsable de mantener los sistemas de bases de datos dentro de la universidad, decidió utilizar un modelo relacional-objeto sobre Oracle para la implementación de la base de datos.

La UT tiene ocho campus en distintas partes del país. La tabla del Campus está enlazada a las tablas de Edificios y Personas. Aunque cada campus ofrece cursos de distintos grados y tiene distintas facultades, en esta etapa, no existe un enlace directo de estos datos con la tabla de Campus.

Campus				
Campus Location	Campus Address	Campus Phone	Campus Fax	Campus Head
Albury/Wodonga	Parkers Road Wodonga VIC 3690	61260583700	620260583777	John Hill
City	215 Franklin St. Melb VIC 3000	61392855100	6103 92855111	Michael A. O'Leary
Mildura	Benetook Ave. Mildura VIC 3502	61350223757	61350223646	Ron Broadhead

La UT tiene cinco facultades, cada una es una agregación de un departamento distinto, escuela y centro de investigación. Cada una debe implementarse como objetos separados. No es necesario acceder de manera independiente los datos de los departamentos, escuelas y centros de investigación en este sistema, se debe usar tablas anidadas.

Faculty					
Fac_ID	Fac_Name	Fac_Deans	Department	School	Research_Centre
1	Health Sciences	S. Duckett			
2	Humanity & Social Sc.	J. A. Salmond			
3	Law & Management	G. C. O'Brien			
4	Science, Tech. & Eng.	D. Finlay			
5	Regional Department	L. Kilmartin			

School			
School_ID	School_Name	School_Head	School_Prof
1-1	Human Biosciences	Chris Handley	Chris Handley
1-2	Human Comm. Sciences	Elizabeth Lavender	Sheena Reilly, Alison Perry, Jan Branson

Department			
Dept_ID	Dept_Name	Dept_Head	Dept_Prof
4-1	Agricultural Sciences	Mark Sandeman	
4-2	Biochemistry	Nick Hoogenraad	Nick Hoogenraad, Robin Anders, Claude Bernard, Bruce Stone

Research_Centre			
RC_ID	RC_Name	RC_Head	RC_Unit
1-1	Australian Research Centre in Sex, Health & Society	Marian Pitts	SSAY Projects HIV Futures Australian Study of Health and Relationships
1-2	Australian Institute for Primary Care	Hal Swerissen	Centre for Dev. and Innovation in Health Centre for Quality in Health & Community Svc. Lincoln Gerontology Centre

Los datos anteriores corresponden a Facultades, Escuelas, Departamentos y Centros de Investigación. Los datos de School_Prof y Dept_Prof corresponden a entidades en sí mismas. Por lo tanto, tendrán sus propios atributos, incluso por ejemplo nombre y contacto.

El atributo Unit del Centro de Investigación tendrá más de un valor, es decir, varias unidades, esta debe implementarse con tipos collection.

Cada campus tiene varios edificios, cada uno de los cuales es una agregación de distintas habitaciones tales como oficinas, salas de clase y laboratorios. La facultad puede ocupar varios edificios. Sin embargo, un edificio sólo puede ser asignado a una facultad. Notese que existe un atributo bld_location, que es la ubicación del edificio en el mapa particular del campus.

Building					
Bld_ID	Bld_Name	Bld_Location	Bld_Level	Campus_Location	Fac_ID
BB1	Beth Gleeson	D5	4	Bundoora	4
BB2	Martin Building	F5	4	Bundoora	3
BB3	Thomas Cherry	D4	4	Bundoora	1
BB4	Physical Science 1	D5	3	Bundoora	4

Como se mencionó antes, un edificio se puede dividir en oficinas, salas de clase y laboratorios, cada uno con sus propios atributos.

Office		
Bld_ID	Off_No	Off_Phone
BB4	BG207	94791118
BB4	BS208	94792393

Classroom		
Bld_ID	Class_No	Class_Capacity
BB3	TCLT	50
BB3	TC01	30

Lab			
Bld_ID	Lab_No	Lab_Capacity	Lab_Equipment
BB1	BG113	25	25 PC, 1 Printer
BB1	BG114	20	21 PC

Nótese que el atributo lab_equipment en Labs puede tener varios valores, debe implementarse usando tipos collection. Para esta agregación se requiere usar la técnica de clustering en vez de tablas anidadas, tomar en cuenta que pueden existir relaciones de asociación entre una Office y otra del Staff que ocupa la oficina.

Cada facultad ofrece a los estudiantes una cantidad de grados. La información sobre los grados se almacena en la tabla Degree. Un grado particular sólo puede ser ofrecido por una facultad particular.

Degree				
Deg_ID	Deg_Name	Deg_Length	Deg_Prereq	Fac_ID
D100	Bachelor of Comp. Sci	3	Year 12 or equivalent	4
D101	Master of Comp. Sci	2	Bach of Comp. Sci	4

Una parte sustancial de la base de datos son los datos del personal. El personal de la universidad se puede categorizar en dos tipos principales: staff y estudiantes. El staff se puede categorizar más en administrador, técnico, profesor (lecturer) y tutor. Un lecturer puede además ser categorizado en senior lecturer y associate lecturer. Un tutor, por otra parte, puede también ser un estudiante, y por lo tanto, debe ser implementado como una relación de herencia múltiple.

Person							
Pers_ID	Pers_Surname	Pers_Fname	Pers_Title	Pers_Address	Pers_Phone	Pers_Postcode	Campus_Location
01234234	Grant	Felix	Mr	2 Boadle Rd Bundoora VIC	0398548753	3083	Bundoora
10008895	Xin	Harry	Mr	6 Kelley St Kew VIC	0398875542	3088	Bundoora
10002935	Jones	Felicity	Ms	14 Rennie St Thornbury VIC	0398722001	3071	Bundoora

Staff			
Pers_ID	Bld_ID	Off_No	Staff_Type
10008895	BB1	BG212	Lecturer
10002935	BB4	BG210	Admin

Student	
Pers_ID	Year
01234234	2000
01958652	2000

Admin			
Pers_ID	Admin_Title	Comp_Skills	Office_Skills
10002935	Office Manager		Managerial
10008957	Receptionist	MS Office	Customer Service, Phone

Technician		
Pers_ID	Tech_Title	Tech_Skills
10005825	Network Officer	UNIX, NT
10015826	Photocopy Technician	Electrician

Lecturer		
Pers_ID	Area	Lect_Type
10008895	Software Engineering	Associate
10000255	Business Information	Senior

Senior Lecturer			
Pers_ID	No_PhD	No_Master	No_Honours
10000255	2	5	7
10000258		1	5

Associate Lecturer		
Pers_ID	No_Honours	Year_Join
10008895	2	1999
10006935		2001

Tutor		
Pers_ID	No_Hours	Rate
01234234	10	20.00
01958652	30	35.00

Los campos vacíos en las muestras anteriores indican que se permiten valores nulos en los atributos correspondientes.

Un estudiante está enlazado con un degree (carrera). Un estudiante puede llevar más de un degree a la vez. El estudiante también está enlazado con Subject (materia). Esta contiene la información sobre la materia: ID, nombre, créditos, prerequisites y descripción. Por otra parte, la materia está enlazada con Lecturer, quien representa el lecturer a cargo de la materia.

Subject				
Subj_ID	Subj_Name	Subj_Credit	Subj_Prereq	Pers_ID
CSE21NET	Networking	10	CSE11IS	10008895
CSE42ADB	Advanced Database	15	CSE21DB	10006935

Enrolls_In es la asociación entre Estudiante y Degree, y Takes es la asociación entre Estudiante y Materia. Utilice referencias a los objetos.

Enrolls_In		Takes		
Student	Degree	Student	Subject	Marks
01234234	D101	01234234	CSE42ADB	70
10012568	D101	10012568	CSE42ADB	80

Estas relaciones deben implementarse como referencias a objetos, no como llave foránea.

STI requiere que se implementen los métodos genéricos para entidades que requerirán muchas actualizaciones, en la base de datos. Estos incluyen Materia, Grado, y todas las relacionadas con personas. También habrán procedimientos almacenados para inserción y borrado en tablas que no derivan de objetos, como Enrolls_In y Takes.

Además de los métodos genéricos, hay algunas consultas definidas por usuarios que se harán frecuentemente para la base de datos. Estas consultas serán implementadas como métodos definidos por usuario:

- Mostrar los nombres y los responsables de escuelas, departamentos y centros de investigación de una facultad. Esta será implementado en Facultad.
- Insertar los datos de un edificio en una nueva tabla, Building_Details. Este será implementado en Edificio.
- Mostrar los detalles de las oficinas y sus ocupantes. Este será implementado en Oficina.
- Guardar en una nueva tabla, Degree_Records, que almacenará los detalles del Degree y la cantidad de estudiantes enrolados en este. Este se implementará en Degree.
- Mostrar los detalles del lecturer que será implementado en Lecturer.

Solución del Problema

Fase 1

Deben implementar todo lo relacionado con la base de datos y el modelo relacional-objeto para el caso de uso, y la lógica de negocios (model y controller en MVC). Deben generar el diagrama de base de datos del modelo relacional-objeto utilizando DataModeler de Oracle. Para cada clase se requiere implementar CRUD (generic methods) y funcionalidades (user defined methods) para manipular sus relaciones con otras clases. Todas estas deberán ser expuestas como una capa de lógica de negocios.

Se requiere que suban un documento PDF con las siguientes partes:

- Portada
- Introducción (0.25%)
- Breve descripción del Problema en Fase 1 (0.25%)
- Diagramas relacional-objeto (2%)
- Diccionario de datos (2%)
- Conclusiones (0.5%)
- Scripts (2.5%)

Todos los scripts utilizados: generación de la base de datos, las agregaciones, tipos (constructores, método de comparación, carga de datos), consultas, procedimientos almacenados, etc. El documento debe ser fácilmente legible, los scripts deben mostrarse en formato fácilmente comprensible.

La Fase 1 tendrá un valor de 7.5%.

Fase 2

Deben implementar dos interfaces para la aplicación: un Web Service RESTful y una aplicación Web que accedan a la lógica de negocios implementada en la primera fase (view en MVC).

Se requiere que suban un ZIP cuyo contenido sea: *los fuentes de las aplicaciones* y un documento en formato PDF con las siguientes partes:

- Portada
- Introducción (0.1%)
- Breve descripción del Problema en Fase 2 (0.2%)
- Documentación de implementación del Web Service RESTful (1%)
- Documentación de implementación de la Aplicación Web (1%)
- Conclusiones (0.2%)

La Fase 2 tendrá un valor de 7.5%: la documentación 2.5% y la funcionalidad completa de las aplicaciones 5% (este cinco por ciento se dividirá por igual entre las distintas funcionalidades a implementar).

Este proyecto se realizará en grupo (los ya creados). Las fechas de entrega para cada fase son las indicadas en el Tec Digital.