

Especificaciones M2 PLANET WARS

La aplicación ha de ser capaz de conectarse a una BBDD Oracle con el fin de poder almacenar los datos de manera persistente.

Para ello, las partes a evaluar del proyecto serán 3.

1. Diseño del modelo de datos a implementar.
2. Scripts
3. Uso de la base de datos.

DISEÑO DEL MODELO DE DATOS A IMPLEMENTAR

Como estamos hablando de un videojuego, es lógico pensar que nuestra aplicación ha de ser capaz de almacenar distinta información dentro de algún lugar, para poder parar y reanudar la partida cuando queramos. Para ello vamos a crear una Base de Datos que será accesible desde la aplicación con las siguientes especificaciones.

- La aplicación ha de ser capaz de almacenar los datos de los usuarios, y para ello tendremos que almacenar todo en una tabla a la que nombraremos USERS, y que deberá almacenar una ID_User que será un campo numérico y PK, un username que será un alfanumérico único para cada usuario, un campo birth_date en formato DD-MM-YYYY y una campo password.
- Un usuario, tendrá que ser capaz, en el caso de que se llegara a implementar en un futuro, de tener varios planetas a su disposición para ir variando de uno a otro. Estos planetas tendrán un ID_Planet, que será numérico y PK, un nombre de planeta que será un alfanumérico único para cada uno de los planetas. Tendremos el valor de la tecnología del ataque (technology_attack) y de la defensa (technology_defense), el coste que tendrá actualizar cada uno de estas tecnologías, así como la cantidad de cristal, metal y deuterium que tenemos en nuestro planeta.
- Se tiene que almacenar el tipo de naves que hay y los tipos de defensas, por separado. A continuación se detallan los campos y valores.

Id_Ship	Name	Metal_Cost	Crystal_Cost	Deuterium_Cost	InitialArmor	Armor	BaseDamage	Speed	GenerateWastings
1	Light Hunter	3000	0	50	400		80	3	20
2	Heavy Hunter	6500	0	50	1000		150	7	30
3	Battle Ship	45000	0	7000	6000		1000	45	60
4	Armored Ship	30000	0	15000	8000		700	70	75
Id_Defense	Name	Metal_Cost	Crystal_Cost	Deuterium_Cost	InitialArmor	Armor	BaseDamage	Speed	GenerateWastings
1	Missile Launcher	2000	0	0	200		80	5	10
2	Ion Cannon	4000	0	500	1200		250	12	25
3	PLasma Cannon	50000	0	5000	7000		2000	30	50

- Se tiene que registrar que naves o defensas tiene cada planeta.
- Se tiene que llevar un registro de las batallas que realiza el usuario, donde solo almacenaremos el ID_Battle y el ID_User, donde una batalla solo puede ser realizada por el usuario. Además se almacenarán las unidades iniciales con las que empezó tanto el usuario como el enemigo, así como las unidades al finalizar esta.

Scripts

La propia aplicación cuando se ejecute será la encargada de inicializar la base de datos, así como de cargar la información necesaria para el funcionamiento de esta. Para ello tendremos los siguientes scripts.

- **01_DROP_TABLE.sql**: Este script borra las tablas siempre que existan. Para realizar el borrado, tenemos dos opciones, borrar las tablas en el orden correcto, o desactivar las restricciones de la base de datos y borrar.
- **02_CREATE_TABLE.sql**: Creará las tablas de la base de datos.
- **03_INSERT_DATA.sql**: Se cargarán los datos necesarios para jugar, en este caso las naves y las defensas.
- **04_INITIALIZE.sql**: Este script, se ejecutará cuando arranque la aplicación, si es la primera vez, que el detecta que no hay tablas creadas, ejecutará los procedures 01 , 02 y 03 en este orden. Si el script detecta que ya están las tablas creadas, no realizará nada. Además, la aplicación dispondrá de un botón “Reset”, que nos devolverá la base de datos a un estado de creación inicial, ejecutando este script y dejando la base de datos limpia. Para ello, es recomendable utilizar flags de true or false, 0 ... 1 ... para poder realizar condiciones de ejecución del o los procedimientos creados.
- **05_GET_DEFENSE.sql - 06_GET_SHIP.sql**: Estos scripts contendrán los procedures necesarios para cargar los datos de las naves y defensas en las variables java necesarias, para inicializar las clases con los setters de las clases ships y defenses de la aplicación. Esta información se recogerá de la base de datos.

USO DE LA BASE DE DATOS.

La aplicación ha de ser capaz de leer y almacenar datos de la misma. Para el uso de la BBDD desde java, se proporcionará código que los alumnos deberán de adaptar y modificar según convenga.

TABLA DE CORRECCIÓN

APARTADO	DESCRIPCIÓN	
DISEÑO DE LA BASE DE DATOS	Diagrama de Chen concorde a las especificaciones dadas.	1
	Modelo relacional concorde al diagrama de chen	1
	Se cumplen especificaciones y restricciones de modelo	1
Scripts	01_DROP_TABLE	0,5
	02_CREATE_TABLE	0,5
	03_INSERT_DATA	0,5
	04_INITIALIZE	0,5
	05_GET_SHIPS	0,5
	06_GET_DEFENSES	0,5
	Control de excepciones	0,5
	Procedure 04 relanzable	0.5
Funcionamiento BBDD	Se cargan datos desde la BBDD	1
	Se guardan datos en la BBDD	1
	Es capaz la aplicación de dejar la BBDD inicializada	1
TOTAL		10