

Entrega 2 Grupo 41

Tablas:

Socios:

<u>id_socio</u>	nombre	nacionalidad

Proyectos:

<u>id_proyecto</u>	tipo	nombre	latitud	longitud	fecha_apertura	operativo

UbicaciónProyecto (Relación):

<u>id_proyecto</u>	latitud	longitud	<u>comuna</u>

LugarProyecto:

<u>comuna</u>	región

Minería:

<u>id_proyecto</u>	mineral

Energía:

<u>id_proyecto</u>	tipo_generación

Participación (Relación):

<u>id_socio</u>	id_proyecto
-----------------	-------------

Recursos:

<u>id_recurso</u>	causa	descripcion	fecha_recurso	comuna_tramite	estatus_limite

UbicaciónTramite(Relación):

<u>id_recurso</u>	comuna_tramite

LugarTramite:

comuna_tramite	region_tramite

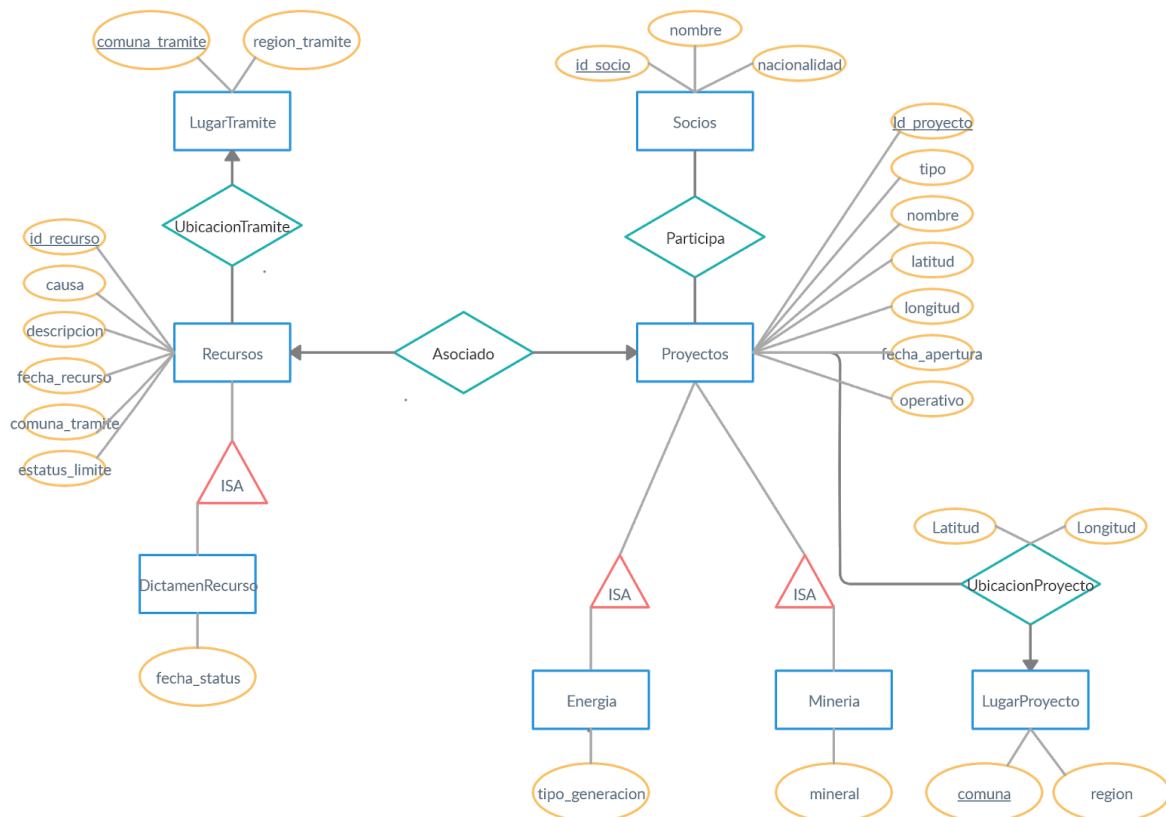
DictamenRecurso:

<u>id_recurso</u>	fecha_status

Asociado(Relación):

<u>id_recurso</u>	id_proyecto

Esquema:



Observación:

- Por falta de tiempo no se pudo cambiar, pero hay que cambiar la cardinalidad.
 - Recursos con LugarTramite es 1:1
 - Recursos Asociado a Proyectos es N:1
 - Socios Participa en Proyectos es N:1
 - Proyectos con LugarProyecto es 1:1

Normalización:

El modelo se encontraba desordenado y con tres tablas muy grandes. Fuimos revisando tabla por tabla, revisando si existían subtablas. Cada subtabla la sacamos de la tabla grande y las fuimos separando y relacionando hasta que en todas las dependencias funcionales no triviales $X \rightarrow Y$, tuviéramos que X es llave o Y es Parte de una llave minimal. Así, más un poquito de redundancia como en la tabla de UbicaciónProyecto (con la latitud y longitud), nuestro modelo quedó 3NF.

Consultas Web:

1. `SELECT * FROM proyectos WHERE tipo='$var';`
2. `SELECT * FROM proyectos, ubicacionproyecto, lugarproyecto WHERE proyectos.id_proyecto = ubicacionproyecto.id_proyectos AND`

ubicacionproyecto.comuna = lugarproyecto.comuna AND lugarproyecto.region = '\$var' AND Proyectos.tipo = 'vertedero';

3. SELECT recursos.id_recurso, recursos.causa, recursos.fecha_recurso, proyectos.tipo FROM recursos, asociado, proyectos WHERE recursos.id_recurso = asociado.id_recurso AND asociado.id_proyectos = proyectos.id_proyecto AND proyectos.tipo = '\$tipo' AND recursos.fecha_recurso > '\$inf' AND recursos.fecha_recurso < '\$sup';
4. SELECT recursos.id_recurso, recursos.causa, recursos.comuna_tramite, lugartramite.region_tramite, recursos.estatus_limite FROM recursos, ubicaciontramite, lugartramite WHERE recursos.id_recurso = ubicaciontramite.id_recurso AND ubicaciontramite.comuna_tramite = lugartramite.comuna_tramite AND (recursos.estatus_limite = 'aprobado' OR recursos.estatus_limite = 'en trámite') ORDER BY lugartramite.region_tramite;
5. SELECT socios.nombre, proyectos.id_proyecto, proyectos.nombre, COUNT(asociado.id_recurso) FROM socios, participa, proyectos, asociado WHERE socios.id_socio = participa.id_socio AND participa.id_proyectos = proyectos.id_proyecto AND proyectos.id_proyecto = asociado.id_proyectos GROUP BY socios.nombre, proyectos.id_proyecto, proyectos.nombre ORDER BY socios.nombre, COUNT(asociado.id_recurso) DESC;
6. SELECT proyectos.id_proyecto, proyectos.tipo, proyectos.nombre, proyectos.operativo, recursos.id_recurso, recursos.estatus_limite FROM proyectos, asociado, recursos WHERE proyectos.id_proyecto = asociado.id_proyectos AND asociado.id_recurso = recursos.id_recurso AND proyectos.operativo = 'si' AND recursos.estatus_limite = 'aprobado' ORDER BY proyectos.id_proyecto;