Gestión de Bases de Datos

Erik Regla, Fabian Olivares

Etapa 1

Desnormalización

Relación 1-1: Se separa columna `plain_text_content` de `problems` porque es muy grande y no se consulta frecuentemente.

Relación 1-N clave: Columna `country_id` añadida a tabla `teams` dado que es muy frecuente consultar el país atribuido a un equipo.

Relación N-M: Nueva columna `team_name` a tabla `team_members` ya que es frecuente consultar los equipos en los que ha participado un competidor.

Tabla de búsqueda: Se separa al malescrito `ballon_colour` de la tabla `problems` ya que son 20 colores que se utilizan. De este modo, reducir espacio en la base de datos.

Restricciones de Dominio

- NOT NULL

```
CREATE TABLE IF NOT EXISTS contestants (
  id SERIAL NOT NULL PRIMARY KEY,
  first_name TEXT NOT NULL,
  last_name TEXT NOT NULL,
  birthdate DATE NOT NULL,
  affiliation TEXT NOT NULL,
  title TEXT NULL,
  sex INTEGER NULL,
  shirt_size TEXT NULL,
  badge_name TEXT NULL,
  home_country INTEGER NOT NULL REFERENCES countries(id) ON DELETE RESTRICT,
  home_city TEXT NOT NULL,
  home_state TEXT NULL,
  ocuppation TEXT NULL,
  special_needs TEXT NULL,
  acm_id INT NULL,
  certificate_name TEXT NULL,
  study_area TEXT NULL,
  degree_pursued TEXT NULL,
  bachelor_start_date DATE NULL,
  bachelor_end_date DATE NULL
);

CREATE TABLE IF NOT EXISTS problems (
  id SERIAL NOT NULL PRIMARY KEY,
  letter TEXT NOT NULL,
  pdf_file TEXT NOT NULL,
  ballon_colour TEXT NOT NULL,
  plain_text_content TEXT NOT NULL,
  description TEXT NOT NULL,
  codename TEXT NOT NULL
);
```

- DEFAULT
  - Secuencias

```
CREATE SEQUENCE contestants_id_seq
    INCREMENT 1
    MINVALUE 1
    MAXVALUE 9223372036854775807
    START 880000
    CACHE 1;
ALTER TABLE contestants_id_seq OWNER TO postgres;
ALTER TABLE contestants ALTER COLUMN id SET DEFAULT nextval('contestants_id_seq');

CREATE SEQUENCE contests_id_seq
    INCREMENT 1
    MINVALUE 1
    MAXVALUE 9223372036854775807
    START 440000
    CACHE 1;
ALTER TABLE contests_id_seq OWNER TO postgres;
ALTER TABLE contests ALTER COLUMN id SET DEFAULT nextval('contests_id_seq');

CREATE SEQUENCE countries_id_seq
    INCREMENT 1
    MINVALUE 1
    MAXVALUE 9223372036854775807
    START 33000
    CACHE 1;
ALTER TABLE countries_id_seq OWNER TO postgres;
ALTER TABLE countries ALTER COLUMN id SET DEFAULT nextval('countries_id_seq');

CREATE SEQUENCE problem_set_id_seq
    INCREMENT 1
```

- CHECK
  - baloon_type_check
- CREATE DOMAIN
  - baloon_type

```
CREATE DOMAIN baloon_type AS text
CONSTRAINT baloon_type_check CHECK (VALUE ~ '([A-Za-z]+( )*)+');
ALTER TABLE colours ALTER COLUMN name SET DATA TYPE baloon_type;
```

Claves y Restricciones de Integridad

- Primary Keys

```
-- primary keys
ALTER TABLE ADD CONSTRAINT colours_pk PRIMARY KEY (id);
ALTER TABLE ADD CONSTRAINT contest_sites_pk PRIMARY KEY (id);
ALTER TABLE ADD CONSTRAINT contestants_pk PRIMARY KEY (id);
ALTER TABLE ADD CONSTRAINT contests_pk PRIMARY KEY (id);
ALTER TABLE ADD CONSTRAINT countries_pk PRIMARY KEY (id);
ALTER TABLE ADD CONSTRAINT problem_set_pk PRIMARY KEY (id);
ALTER TABLE ADD CONSTRAINT problems_pk PRIMARY KEY (id);
ALTER TABLE ADD CONSTRAINT problems_content_pk PRIMARY KEY (id);
ALTER TABLE ADD CONSTRAINT roles_pk PRIMARY KEY (id);
ALTER TABLE ADD CONSTRAINT scoreboards_pk PRIMARY KEY (id);
ALTER TABLE ADD CONSTRAINT sites_pk PRIMARY KEY (id);
ALTER TABLE ADD CONSTRAINT team_members_pk PRIMARY KEY (id);
ALTER TABLE ADD CONSTRAINT teams_pk PRIMARY KEY (id);
```

- Foreign Keys y Restricciones

```
-- foreign keys
ALTER TABLE contest_sites ADD FOREIGN KEY (id_site) REFERENCES sites ON DELETE CASCADE ON UPDATE CASCADE;
ALTER TABLE contest_sites ADD FOREIGN KEY (id_contest) REFERENCES contests ON DELETE CASCADE ON UPDATE CASCADE;
ALTER TABLE contestants ADD FOREIGN KEY (home_country) REFERENCES countries ON DELETE SET NULL ON UPDATE CASCADE;
ALTER TABLE problem_set ADD FOREIGN KEY (contest_id) REFERENCES contests ON DELETE CASCADE ON UPDATE CASCADE;
ALTER TABLE problem_set ADD FOREIGN KEY (problem_id) REFERENCES problem ON DELETE RESTRICT ON UPDATE CASCADE;
ALTER TABLE problems ADD FOREIGN KEY (colour_id) REFERENCES colour ON DELETE SET NULL ON UPDATE CASCADE;
ALTER TABLE scoreboards ADD FOREIGN KEY (problem_id) REFERENCES problems ON DELETE RESTRICT ON UPDATE CASCADE; --transaction
ALTER TABLE scoreboards ADD FOREIGN KEY (contest_id) REFERENCES contests ON DELETE RESTRICT ON UPDATE CASCADE; --transaction
ALTER TABLE scoreboards ADD FOREIGN KEY (team_id) REFERENCES teams ON DELETE RESTRICT ON UPDATE CASCADE; --transaction
ALTER TABLE sites ADD FOREIGN KEY (country_id) REFERENCES countries ON DELETE SET NULL ON UPDATE CASCADE; --transaction
ALTER TABLE team_members ADD FOREIGN KEY (contestant_id) REFERENCES contestants ON DELETE RESTRICT ON UPDATE RESTRICT; --transaction
ALTER TABLE team_members ADD FOREIGN KEY (role_id) REFERENCES roles ON DELETE SET NULL ON UPDATE CASCADE; --transaction
ALTER TABLE team_members ADD FOREIGN KEY (team_id) REFERENCES teams ON DELETE RESTRICT ON UPDATE CASCADE; --transaction
ALTER TABLE teams ADD FOREIGN KEY (coach_id) REFERENCES contestants ON DELETE SET NULL ON UPDATE CASCADE;
ALTER TABLE teams ADD FOREIGN KEY (site_id) REFERENCES sites ON DELETE CASCADE ON UPDATE CASCADE;
ALTER TABLE teams ADD FOREIGN KEY (country_id) REFERENCES countries ON DELETE SET NULL ON UPDATE CASCADE; --transaction
```

Índices

```
-- PROBLEM INDICES
CREATE INDEX problem_id_idx ON problems USING hash (id);
CREATE INDEX problem_letter_idx ON problems USING hash (letter);
CREATE INDEX problem_pdf_file_idx ON problems USING hash (pdf_file);
CREATE INDEX problem_colour_id_idx ON problems USING hash (colour_id);
CREATE INDEX problem_description_id_idx ON problems USING hash (description);
CREATE INDEX problem_codename_idx ON problems USING hash (codename);

-- Problem content :p
CREATE INDEX problems_content_id_idx ON problems_content USING hash (id);
--CREATE INDEX problems_content_plain_text_content_idx ON problems_content USING hash (plain_text_content);

-- Sites
CREATE INDEX sites_institution_idx ON sites USING hash (institution);

-- Team members
CREATE INDEX team_members_id_idx ON team_members USING hash (id);
CREATE INDEX team_members_contestant_id_idx ON team_members USING hash (contestant_id);
CREATE INDEX team_members_team_id_idx ON team_members USING hash (team_id);
CREATE INDEX team_members_role_id_idx ON team_members USING hash (role_id);
CREATE INDEX team_members_registration_complete_idx ON team_members USING hash (registration_complete);
CREATE INDEX team_members_on_team_certificate_idx ON team_members USING hash (on_team_certificate);
CREATE INDEX team_members_on_individual_certificate_idx ON team_members USING hash (on_individual_certificate);
CREATE INDEX team_members_team_name_idx ON team_members USING hash (team_name);

-- TEAMS!
CREATE INDEX team_id_idx ON teams USING hash (id);
CREATE INDEX team_institution_idx ON teams USING hash (institution);
CREATE INDEX team_coach_id_idx ON teams USING hash (coach_id);
CREATE INDEX team_name_idx ON teams USING hash (name);
CREATE INDEX team_site_id_idx ON teams USING hash (site_id);
CREATE INDEX team_approved_idx ON teams USING hash (approved);
CREATE INDEX team_include_coach_cert_idx ON teams USING hash (include_coach_cert);
CREATE INDEX team_make_coach_individual_cert_idx ON teams USING hash (make_coach_individual_cert);
CREATE INDEX team_country_id_idx ON teams USING hash (country_id);
```

Reglas de Negocio

- No se puede repetir la letra de un problema en un mismo problem set.
- No se puede repetir el color del globo en un mismo problem set.
- Un mismo problema no se puede asignar más de una vez al mismo contest.
- Al borrar un problema, se borran las entradas asociadas a éste en el scoreboard.
- Un team debe estar compuesto por exactamente tres contestants.
- Un contestant asociado a un team debe pertenecer a la misma institución que dicho team.

Etapa 2

Transacciones

- **Actualización o cambio de problemas:**

  Cuando se efectúa un cambio en un problema, por reglamento, todos los puntajes asociados a ese problema deben ser eliminados. De no ser así, el problema no puede cambiar de estado. En caso de ser interrumpido el proceso requiere de **savepoints** para poder reanudar o bien abortar la operación al vuelo.

- **Creación de set de problemas:**

  Para crear un set de problemas, es necesario incluirlos todos y este problem set debe estar ligado a una competencia. No puede existir un problem set sin competencia ni un problema que no pertenezca a un set. De fallar alguno de estos pasos, es necesario revertir los cambios para evitar lanzar una competencia con datos inválidos.

- **Creación de equipos:**

  Para crear correctamente un equipo, se debe crear la instancia y además conectar a los participantes. En caso de fallar uno, todo debe ser deshecho.

FIN