**SQL**

**PROCEDURE insertNormalUser(email VARCHAR(50), pass VARCHAR(25))** : Inserta el usuario email con la contraseña pass en la tabla User y crea su perfil Profile y su perfil Partier vacio.

**PROCEDURE insertFacebookUser(email VARCHAR(50)) :** Inserta el usuario email en la tabla User y crea su perfil Profile y su perfil Partier vacio

**PROCEDURE insertGoogleUser(email VARCHAR(50)) :** Inserta el usuario email en la tabla User y crea su perfil Profile y su perfil Partier vacio

**FUNCTION userExists(email VARCHAR(50)) RETURNS BOOLEAN:** Devuelve 1 si el usuario email existe, 0 en caso contrario.

**FUNCTION passOK(usr VARCHAR(50), pass VARCHAR(25)) RETURNS BOOLEAN :** Devuelve 1 si la contraseña asociada al usuario usr es pass y su tipo es usuario normal.

**PROCEDURE deleteToken(IN email VARCHAR(50)) :** Pone a null el campo token del usuario email en la tabla User.

**FUNCTION tokenEquals(usr VARCHAR(50), tkn VARCHAR(45)) RETURNS BOOLEAN** : Devuelve 1 si tkn coincide con el campo token asociado al email usr de la tabla User, 0 en caso contrario.

**PROCEDURE updateToken(IN email VARCHAR(50), IN tkn VARCHAR(45)) :** Actualiza el campo token asociado a email en la tabla user por tkn.

**FUNCTION getIdProfile(usr VARCHAR(50)) RETURNS INT :** Devuelve el idProfile asociado al usuario usr en la tabla Profile.

**PROCEDURE setPartierData(IN idProfile INT, IN picture VARCHAR(100), IN `name`VARCHAR(20), IN surnames VARCHAR(45), IN birthdate DATE, IN gender BOOL, IN music VARCHAR(20), IN civil\_state VARCHAR(20), IN city VARCHAR(20), IN drink VARCHAR(20), IN about VARCHAR(200)) :** Actualiza todos los campos de Partier con los datos introducidos en la columna que coincida con idProfile.

**PROCEDURE getPartierData(IN idProfile INT) :** Devuelve todos los datos del Partier cuyo idProfile coincida con el introducido.