

# Indexes, Triggers, User Functions and Population - Artifact 6

This artefact contains the physical schema of the database, the database load, the identification and characterization of the indexes, triggers to enforce integrity rules that can not be achieved in a simpler way and the definition of the database user-defined functions. This artefact also includes the complete database creation script, including all SQL necessary to define all integrity constraints, indexes and triggers.

## Database Load

Relation		Tuple's Order of Magnitude	Estimated Growth per Day
R01	User	1 000	The estimated growth per day of the User table is 5 users per day.
R02	Sports Complex	100	The estimated growth per day of the SportsComplex table is 0.5 sport complex per day.
R03	Space	1 000	The estimated growth per day of the Space table is 1 space per day.
R04	Equipment	1 000	The estimated growth per day of the Equipment table is 5 new equipment per day.
R05	Rental	10 000	The estimated growth per day of the Rental table is 10 rentals per day.
R06	Issue	100	The estimated growth per day of the Issue table is 0.5 issues per day.
R07	Sport	10	This table will not have any growth per day.
R08	Municipality	10	This table will not have any growth per day.

## Important queries (SELECT) and their frequency

<b>SELECT01</b>	List of sports complexes chosen by name, municipality, sport, date of rental, start time and duration of rental, surface and coverage of the spaces. Any of these fields can be null. In that case they don't contribute to the search and complexes are selected according to the other fields. Calls to this SELECT will be frequent. This list will be the result of the advanced search placed in the "search results" page of the web application. The variables \$complexName, \$complexMunicipality, \$rentalStartTime, \$rentalDuration, \$rentalDate, \$surfaceType, \$spacesCovered come from the filled form.
-----------------	--

```

SELECT "complexID", "complexName", "municipalityName", "complexPhone", "complexEmail"
FROM "SportsComplex"
JOIN "Municipality" ON "complexMunicipalityID" = "municipalityID"
WHERE
/* NAME */
(to_tsvector('english', "complexName") @@ plainto_tsquery($complexName) OR ($complexName IS
NULL))
AND
/* MUNICIPALITY */
("municipalityID" = $complexMunicipality OR ($complexMunicipality IS NULL))
AND
/* OPENING TIME */
( $rentalStartTime >= "complexOpeningHour" OR $rentalStartTime IS NULL)
AND
/* CLOSING TIME */
((( $rentalStartTime+ $rentalDuration) <= "complexClosingHour") OR $rentalStartTime IS NULL OR
$rentalDuration IS NULL)
AND
(/* WEEKEND */
"complexOpenOnWeekends" = TRUE
OR
(((EXTRACT(DOW FROM $rentalDate + '00:00:00'::TIME) > 0 AND EXTRACT(DOW FROM $rentalDate +
'00:00:00'::TIME) < 6) OR $rentalDate IS NULL)))
AND
/* COMPLEX ACTIVE */
"complexInactive" = FALSE
AND
EXISTS
(SELECT "spaceID"
FROM "Space"
WHERE
"spaceComplexID" = "complexID"
AND
/* SPORT */
EXISTS
(SELECT "sportID"
FROM "Sport"
JOIN "SpaceSports" ON "sportID" = "spaceSportsSportID"
WHERE
"spaceID" = "spaceSportsSpaceID"
AND
("sportID" = $sportID OR $sportID IS NULL))
AND
/* AVAILABILITY */
"spaceIsAvailable" = TRUE
AND
(NOT EXISTS
(SELECT "rentalID"
FROM "Rental"
WHERE
"spaceID" = "rentalSpaceID"
AND
"rentalState" = 'RESERVED'::"rentalState"
AND(
("rentalDate" + "rentalStartTime" = ($rentalDate + $rentalStartTime))
OR
(("rentalDate" + "rentalStartTime" < ($rentalDate + $rentalStartTime))
AND ("rentalDate" + "rentalStartTime" + "rentalDuration" > ($rentalDate +
$rentalStartTime)))
OR
(("rentalDate" + "rentalStartTime" > ($rentalDate + $rentalStartTime))
AND ("rentalDate" + "rentalStartTime" < ($rentalDate + $rentalStartTime +
$rentalDuration))))))
OR
($rentalDate IS NULL OR $rentalStartTime IS NULL OR $rentalDuration IS NULL))
AND
/* SURFACE TYPE */
("spaceSurfaceType" = $surfaceType OR $surfaceType IS NULL)
AND
/* COVERAGE */
("spaceIsCovered" = $spaceIsCovered OR $spaceIsCovered IS NULL))
;

```

<b>SELECT02</b>	<p>List of active spaces of a sports complex - calls to this SELECT will be frequent because this list will be part of the sports complex page, where the user will be able to choose to rent an item.</p> <pre> SELECT "spaceID", "spaceName", "spaceIsCovered", "spaceSurfaceType",       "spacePrice" FROM "Space" WHERE "spaceComplexID" = \$complexID AND "spaceIsAvailable" = true; </pre>
<b>SELECT03</b>	<p>List of rentals of the user logged in the web application ordered by date - calls to this SELECT will be frequent. This list will be displayed in the "manage rentals" page of the web application.</p> <pre> Select "rentalID", "rentalDate", "rentalDuration", "rentalState", "spaceName", "complexName", "rentalStartTime", "rentalRating", "complexID",       "spaceID" From "Rental", "User", "Space", "SportsComplex" Where       "userID" = \$userID       AND "userID" = "rentalUserID"       AND "rentalSpaceID" = "spaceID"       AND "spaceComplexID" = "complexID" ORDER BY "rentalDate", "rentalStartTime" DESC; </pre>
<b>SELECT04</b>	<p>List of sports complexes of a manager (user that is logged in the web application) - calls to this SELECT will be frequent. This list will be displayed in the "manage complexes" page of the web application.</p> <pre> Select "complexName", "complexID" From "SportsComplex", "User", "Manager" Where       "complexID" = "managerComplexID"       AND "managerID" = "userID"       AND "userUsername" = \$userUsername; </pre>
<b>SELECT05</b>	<p>List of rentals of a sports complex ordered by date - calls to this SELECT will be frequent. This list will be displayed in the "manage rentals" page, accessible in the "manage complexes" page.</p> <pre> SELECT "userUsername", "userName", "userPhone", "userEmail", "rentalDate", "spaceName", "rentalStartTime", "rentalDuration", "rentalState", "rentalID",       "spaceID" FROM "Rental"       JOIN "Space" ON "spaceID" = "rentalSpaceID"       JOIN "SportsComplex" ON "complexID" = "spaceComplexID"       JOIN "User" ON "userID" = "rentalUserID" WHERE "complexID" = \$complexID; </pre>
<b>SELECT06</b>	<p>List of equipment of a sports complex - calls to this SELECT will be frequent. This list will be displayed in the "manage equipment" page accessible in the "manage complexes" page.</p>

```

SELECT "equipmentID", "equipmentName", "equipmentQuantity",
      "equipmentPrice",
"equipmentDetails", "equipmentQuantityUnavailable", "equipmentInactive",
      "equipmentSportsSportID"
      FROM "Equipment"
      JOIN "EquipmentSports"
ON "equipmentSportsEquipmentID" = "equipmentID"
WHERE "equipmentComplexID" = $complexID;

```

<b>SELECT07</b>	List of issues of a sports complex ordered by date - calls to this SELECT will be sometimes. This list will be displayed in the "manage issues" page accessible in the "manage complexes" page.
<pre> SELECT "userName", "userPhone", "userEmail", "rentalDate", "rentalState",       "issueID", "issueSubject", "issueCategory", "issueDescription",       "issueResolved", "spaceName", "rentalStartTime"       FROM "Issue"       JOIN "Rental" ON "rentalID" = "issueRentalID"       JOIN "User" ON "rentalUserID" = "userID"       JOIN "Space" ON "spaceID" = "rentalSpaceID"       WHERE       "spaceComplexID" = \$complexID       AND "issueForManager" = TRUE; </pre>	

<b>SELECT08</b>	List of managers of a sports complex - calls to this SELECT will be made sometimes. This list will be displayed in the "manage managers" page accessible in the "manage complexes" page.
<pre> SELECT "managerID", "userName", "userEmail", "userPhone"       FROM "Manager"       JOIN "User" ON "userID" = "managerID"       WHERE "managerComplexID" = \$complexID; </pre>	

<b>SELECT09</b>	List of equipment of a sports complex available for rent at a determined date and time inserted in the space page by the user - calls to this SELECT will be frequent. This information will be displayed at the bottom of the space page with the option to select which equipment and how many of each the user will want to rent.
-----------------	--

```

        SELECT
            "equipmentID"
            ,"equipmentName"
            ,"equipmentQuantity"
            ,"equipmentQuantityUnavailable"
            ,"equipmentPrice"
        , SUM(CASE WHEN "rentalEquipmentQuantity" IS NULL THEN 0 ELSE
            "rentalEquipmentQuantity" END) As "rentalQuantity"
        FROM "Space"
        JOIN "SportsComplex" ON "spaceComplexID" = "complexID"
        JOIN "Equipment" ON "complexID" = "equipmentComplexID"
        JOIN "EquipmentSports" ON "equipmentID" = "equipmentSportsEquipmentID"
        JOIN "Sport" ON "equipmentSportsSportID" = "sportID"
        LEFT JOIN
            (
                SELECT "equipmentID" as "cenas","rentalEquipmentQuantity"
                FROM "Space"
                JOIN "SportsComplex" ON "spaceComplexID" = "complexID"
                JOIN "Equipment" ON "complexID" = "equipmentComplexID"
                JOIN "EquipmentSports" ON "equipmentID" = "equipmentSportsEquipmentID"
                JOIN "Sport" ON "equipmentSportsSportID" = "sportID"
                LEFT JOIN "RentalEquipment" ON "equipmentID" = "rentalEquipmentEquipmentID"
                LEFT JOIN "Rental" ON "rentalEquipmentRentalID" = "rentalID"
                WHERE "spaceID" = $spaceID
                AND "sportName" IN (
                    SELECT "sportName"
                    FROM "SpaceSports"
                )
                JOIN "Sport" ON "sportID" = "spaceSportsSportID"
                WHERE "spaceSportsSpaceID" = $spaceID
            )
            AND "equipmentInactive" = FALSE
            AND "rentalState" = 'RESERVED'::"rentalState"
            AND (
                (
                    ("rentalDate" + "rentalStartTime" = ($rentalDate + $rentalStartTime))
                    OR (
                        ("rentalDate" + "rentalStartTime" < ($rentalDate +
                            $rentalStartTime))
                        AND ("rentalDate" + "rentalStartTime" + "rentalDuration" >
                            ($rentalDate + $rentalStartTime))
                    )
                )
                OR (
                    ("rentalDate" + "rentalStartTime" > ($rentalDate +
                        $rentalStartTime))
                    AND ("rentalDate" + "rentalStartTime" < ($rentalDate +
                        $rentalStartTime + $rentalDuration))
                )
            )
            OR ("rentalEquipmentQuantity" IS NULL)
        ) As "xxx"
        ON "equipmentID" = "cenas"
        WHERE "spaceID" = $spaceID
        AND "sportName" IN (
            SELECT "sportName"
            FROM "SpaceSports"
        )
        JOIN "Sport" ON "sportID" = "spaceSportsSportID"
        WHERE "spaceSportsSpaceID" = $spaceID
        )
        AND "equipmentInactive" = FALSE
        GROUP BY "equipmentID";

```

<b>SELECT10</b>	Information of the sports complex to be showed in the sports complex page - calls to this SELECT will be frequent.
<pre>SELECT * FROM "SportsComplex" WHERE "complexID" = \$complexID;</pre>	

<b>SELECT11</b>	Information of the space to be showed in the space page - calls to this SELECT will be frequent.
<pre>SELECT "spaceName", "spaceIsCovered", "spaceSurfaceType", "spacePrice", "spaceComplexID", "complexName", "complexLocation", "complexEmail", "complexPhone", "complexOpeningHour", "complexClosingHour", "complexOpenOnWeekends" FROM "Space" JOIN "SportsComplex" ON "spaceComplexID" = "complexID" WHERE "spaceID" = \$spaceID;</pre>	

<b>SELECT12</b>	List of spaces of a sports complex - calls to this SELECT will be frequent. This list will be displayed in the "manage spaces" page accessible in the "manage complexes" page.
<pre>SELECT "spaceID", "spaceName", "spaceIsCovered", "spaceSurfaceType", "spacePrice" FROM "Space" WHERE "spaceComplexID" = \$complexID AND "spaceIsAvailable" = true;</pre>	

## Important changes (UPDATE/DELETE) and their frequency

<b>UPDATE01</b>	Update user information - it is not expected for this update to be called frequently. It allows the user to edit his information on the web application through a form where he can introduce the new information used in this query with the variables: \$userName, \$userEmail, \$userUsername, \$userPhone, \$userMunicipalityID and \$userID.
<pre>UPDATE "User" SET "userName" = \$userName, "userEmail" = \$userEmail, "userUsername" = \$userUsername, "userPhone" = \$userPhone, "userMunicipalityID" = \$userMunicipalityID WHERE "userID" = \$userID;</pre>	
<b>UPDATE02</b>	Update sports complex information - it is not expected for this update to be called frequently. It allows the manager of a sports complex to edit it's information on the web application through a form where he can introduce the new information used in this query with the variables: \$complexName, \$complexLocation, \$complexMunicipalityID, \$complexEmail, \$complexPhone, \$complexPaypal, \$complexOpenOnWeekends, \$complexDescription, \$complexOpeningHour, \$complexInactive and \$complexClosingHour.

```

UPDATE "SportsComplex"
SET
    "complexName" = $complexName,
    "complexLocation" = $complexLocation,
    "complexMunicipalityID" = $complexMunicipalityID,
    "complexEmail" = $complexEmail,
    "complexPhone" = $complexPhone,
    "complexDescription" = $complexDescription,
    "complexOpeningHour" = $complexOpeningHour,
    "complexClosingHour" = $complexClosingHours,
    "complexOpenOnWeekends" = $complexOpenOnWeekends,
    "complexPaypal" = $complexPaypal,
    "complexInactive" = $complexInactive
WHERE
    "complexID" = $complexID;

```

<b>UPDATE03</b>	Update space information - it is not expected for this update to be called frequently. It allows the manager of a sports complex to edit a specific space information within the sports complex, through a form where he can introduce the new information used in this query with the variables: \$spaceName, \$spaceIsCovered, \$spaceIsAvailable, \$spacePrice and \$spaceSurfaceType.
-----------------	---

```

UPDATE "Space"
SET
    "spaceName" = $spaceName,
    "spaceSurfaceType" = $spaceSurfaceType,
    "spaceIsCovered" = $spaceIsCovered,
    "spaceIsAvailable" = $spaceIsAvailable,
    "spacePrice" = $spacePrice
WHERE
    "spaceID" = $spaceID;

```

<b>UPDATE04</b>	Update equipment information - it is expected for this update to be called frequently. It allows the manager of a sports complex to edit the information of it's equipment, through a form where he can introduce the new information used in this query with the variables: \$equipmentName, \$equipmentQuantity, \$equipmentDetails, \$equipmentQuantityUnavailable, \$equipmentInactive and \$equipmentPrice.
-----------------	--

```

UPDATE "Equipment"
SET
    "equipmentName" = $equipmentName,
    "equipmentQuantity" = $equipmentQuantity,
    "equipmentDetails" = $equipmentDetails,
    "equipmentPrice" = $equipmentPrice,
    "equipmentQuantityUnavailable" = $equipmentQuantityUnavailable,
    "equipmentInactive" = $equipmentInactive
WHERE
    "equipmentID" = $equipmentID;

```

<b>UPDATE05</b>	Update user password - it is not expected for this update to be called frequently. It allows the user to change his password through a form where he can introduce the new password. It is confirmed if the old password matches through a SELECT before this UPDATE is done. The new password is in the variable \$userPassword.
-----------------	---

```
UPDATE "User"
SET
"userPassword" = $userPassword
WHERE
"userID" = $userID;
```

<b>UPDATE06</b>	Update rental state to "waiting user feedback" - it is expected for this update to be called very frequently. It checks if the rental starting time plus the duration has passed the realtime date and time and updates the rental state according to that information.
<pre>UPDATE "Rental" SET "rentalState" = 'WAITINGUSERFEEDBACK'::"rentalState" WHERE "rentalState" = 'RESERVED'::"rentalState" AND ((now() - (("rentalStartTime" + "rentalDate")) &lt; '24 hours'::interval) AND ((now() - (("rentalStartTime" + "rentalDate")) &gt; '0'::interval);</pre>	
<b>UPDATE07</b>	Cancel rental by the user - it is not expected for this update to be called frequently. It allows the user to cancel of his rentals.
<pre>UPDATE "Rental" SET "rentalState" = 'CANCELEDBYUSER'::"rentalState" WHERE "rentalID" = \$rentalID;</pre>	
<b>UPDATE08</b>	Cancel rental by the manager - it is not expected for this update to be called frequently. It allows the sports complex manager to cancel one of the rentals.
<pre>UPDATE "Rental" SET "rentalState" = 'CANCELEDBYMANAGER'::"rentalState" WHERE "rentalID" = \$rentalID;</pre>	
<b>UPDATE09</b>	Admin suspend rental - it is not expected for this update to be called frequently. It allows the website administrator to suspend one of the rentals.
<pre>UPDATE "Rental" SET "rentalState" = 'SUSPENDEDADMIN'::"rentalState" WHERE "rentalID" = \$rentalID;</pre>	
<b>UPDATE10</b>	Conclude rental - it is expected for this update to be called frequently. It allows the website administrator, manager or user to conclude one of the rentals.
<pre>UPDATE "Rental" SET "rentalState" = 'CONCLUDED'::"rentalState" WHERE "rentalID" = \$rentalID;</pre>	
<b>UPDATE11</b>	Cancel rental by administrator - it is not expected for this update to be called frequently. It allows the website administrator to cancel one of the rentals.
<pre>UPDATE "Rental" SET "rentalState" = 'CANCELEDBYADMIN'::"rentalState" WHERE "rentalID" = \$rentalID;</pre>	



<b>UPDATE11</b>	Cancel rental by administrator - it is not expected for this update to be called frequently. It allows the website administrator to cancel one of the rentals.
<pre>UPDATE "Rental" SET "rentalState" = 'CANCELEDBYADMIN'::"rentalState" WHERE "rentalID" = \$rentalID;</pre>	

## Indexes and Clustering

Here are presented the indexes created for our project. We came to the conclusion that it wouldn't be possible to implement clusters in our database because no tables have one or more columns in common.

Index created for faster selects related to the home page search form for the sports complex, where the search of a sports complex by it's location is a frequent query.

<b>IDX1</b>	sportComplexSearchIndex	complexID, complexLocation	None
<pre>CREATE INDEX "sportComplexSearchIndex" ON "SportsComplex" USING hash ("complexLocation");</pre>			

Index created for faster selects related to the results of the home page search form, when used to filter the sports complex space's by the surface type and coverage.

<b>IDX2</b>	spaceComplexSearchIndex	spaceComplexID, spaceSurfaceType, spaceIsCovered	None
<pre>CREATE INDEX "spaceComplexSearchIndex" ON "Space" USING btree ("spaceComplexID", "spaceSurfaceType", "spaceIsCovered");</pre>			

Index created for faster selects related to the rentals of a user.

<b>IDX3</b>	rentalUserIndex	rentalUserID	None
<pre>CREATE INDEX "rentalUserIndex" ON "Rental" USING hash ("rentalUserID");</pre>			

Index created for faster selects related to the query needed to know if a space is already occupied by a rental when searching for spaces available at a certain time.

<b>IDX4</b>	rentalSpaceTimeIndex	rentalID, rentalSpaceID, rentalStartTime, rentalDurationInMinutes	None
<pre>CREATE INDEX "rentalSpaceTimeIndex" ON "Rental" USING btree ("rentalID", "rentalSpaceID", "rentalStartTime", "rentalDurationInMinutes");</pre>			

Index created for faster selects related to the search for equipment of a sports complex.

<b>IDX5</b>	equipmentInfoIndex	equipmentComplexID, equipmentID, equipmentQuantity, equipmentQuantityUnavailable	None
<pre>CREATE INDEX "equipmentInfoIndex" ON "Equipment" USING btree ("equipmentComplexID", "equipmentID", "equipmentQuantity", "equipmentQuantityUnavailable");</pre>			

Index created for faster selects related to the search for an issue related to a rental.

IDX6	issueRentalIndex	issueID, issueRentalID	None
CREATE INDEX "issueRentalIndex" ON "Issue" USING hash ("issueRentalID");			

## Full-Text Search

The “full-text search” will be of use in the following fields:

Table	Attribute	Type
SportsComplex	complexName	varchar(64)
	complexLocation	varchar(256)
	complexDescription	varchar(256)

For the use of the “full-text search” capability, the creation of the following indexes was needed:

```
CREATE INDEX sportcomplexsearchindexfulltextweighted ON "SportsComplex"
USING gin (((setweight(to_tsvector('english'::regconfig,
"complexName"::text), 'A'::"char") ||
setweight(to_tsvector('english'::regconfig, "complexLocation"::text),
'B'::"char"))) || setweight(to_tsvector('english'::regconfig,
"complexDescription"::text), 'C'::"char")));
```

```
CREATE INDEX sportcomplexsearchindexname ON "SportsComplex" USING gin
(to_tsvector('english'::regconfig, "complexName"::text));
```

After the indexes creation, the search will now be faster and more efficient. In the queries presented, a “full-text search” is done using the indexes sportcomplexsearchindexfulltextweighted and sportcomplexsearchindexname respectively presented above.

This SELECT is used in search form of the website's home page, redirecting to the search results page, where all the results of this search will be presented.

```
SELECT "complexID", "complexName", "complexPhone", "complexEmail",
"complexLocation"
FROM "SportsComplex"
WHERE
setweight(to_tsvector('english', "complexName"), 'A')
|| ' ' ||
setweight(to_tsvector('english', "complexLocation"), 'B')
|| ' ' ||
setweight(to_tsvector('english', "complexDescription"), 'C')
@@ plainto_tsquery($search);
```

This SELECT is used in advanced search form of the website's search results page, where all the results of this search will be presented.

```
SELECT "complexID", "complexName", "municipalityName", "complexPhone",
```

```

"complexEmail"
FROM "SportsComplex"
JOIN "Municipality" ON "complexMunicipalityID" = "municipalityID"
WHERE
/* NAME */
(to_tsvector('english', "complexName") @@ plainto_tsquery($complexName) OR
($complexName IS NULL))
AND
/* MUNICIPALITY */
("municipalityID" = $complexMunicipality OR ($complexMunicipality IS NULL))
AND
/* OPENING TIME */
( $rentalStartTime >= "complexOpeningHour" OR $rentalStartTime IS NULL)
AND
/* CLOSING TIME */
(((($rentalStartTime+ $rentalDuration) <= "complexClosingHour") OR
$rentalStartTime IS NULL OR $rentalDuration IS NULL)
AND
(/* WEEKEND */
"complexOpenOnWeekends" = TRUE
OR
(((EXTRACT(DOW FROM $rentalDate + '00:00:00')::TIME) > 0 AND EXTRACT(DOW
FROM $rentalDate + '00:00:00')::TIME) < 6) OR $rentalDate IS NULL)))
AND
/* COMPLEX ACTIVE */
"complexInactive" = FALSE
AND
EXISTS
(SELECT "spaceID"
FROM "Space"
WHERE
"spaceComplexID" = "complexID"
AND
/* SPORT */
EXISTS
(SELECT "sportID"
FROM "Sport"
JOIN "SpaceSports" ON "sportID" = "spaceSportsSportID"
WHERE
"spaceID" = "spaceSportsSpaceID"
AND
("sportID" = $sportID OR $sportID IS NULL))
AND
/* AVAILABILITY */
"spaceIsAvailable" = TRUE
AND
(NOT EXISTS
(SELECT "rentalID"
FROM "Rental"
WHERE
"spaceID" = "rentalSpaceID"

```

```

        AND
        "rentalState" = 'RESERVED'::"rentalState"
        AND(
            ("rentalDate" + "rentalStartTime" = ($rentalDate +
$rentalStartTime))
            OR
            (("rentalDate" + "rentalStartTime" < ($rentalDate +
$rentalStartTime))
                AND ("rentalDate" + "rentalStartTime" +
"rentalDuration" > ($rentalDate + $rentalStartTime)))
            OR
            (("rentalDate" + "rentalStartTime" > ($rentalDate +
$rentalStartTime))
                AND ("rentalDate" + "rentalStartTime" < ($rentalDate
+ $rentalStartTime + $rentalDuration))))
            OR
            ($rentalDate IS NULL OR $rentalStartTime IS NULL OR
$rentalDuration IS NULL))
        AND
        /* SURFACE TYPE */
        ("spaceSurfaceType" = $surfaceType OR $surfaceType IS NULL)
        AND
        /* COVERAGE */
        ("spaceIsCovered" = $spaceIsCovered OR $spaceIsCovered IS NULL))
    ;

```

## Triggers and User-defined Functions

<b>RN01</b>	This trigger will allow to suspend all the rentals of a determined user when this one is blocked by one of the website's administrator. The following function is used for this Trigger. It allows us to change the rental state according to the state of the user (banner or not banned).
<pre> CREATE TRIGGER banneduser AFTER UPDATE OF "userIsBanned" ON "User" FOR EACH ROW EXECUTE PROCEDURE banneduser();  CREATE FUNCTION banneduser() RETURNS TRIGGER LANGUAGE plpgsql AS \$\$BEGIN IF (OLD."userIsBanned" = FALSE AND NEW."userIsBanned" = TRUE) THEN UPDATE "Rental" SET "rentalState" = 'SUSPENDED BY ADMIN'::"rentalState" WHERE "rentalUserID" = NEW."userID"; END IF; RETURN NEW; END; \$\$; </pre>	

<b>RN02</b>	<p>This trigger will allow to set a reservation state to “concluded” as soon as the issue associated is removed by a manager (when it was resolved or it is not valid anymore). When a reservation as an issue associated, it’s state is set to suspended until the reservation is resolved by the manager or the administrator decides to change it’s state. The following function is used for this Trigger. It allows us to change the rental state according to the state of the issue associated with it.</p>
	<pre>CREATE TRIGGER concludedissue AFTER UPDATE OF "issueResolved" ON "Issue" FOR     EACH ROW EXECUTE PROCEDURE concludedissue();  CREATE FUNCTION concludedissue() RETURNS trigger     LANGUAGE plpgsql     AS \$\$BEGIN     IF(OLD."issueResolved" = false AND New."issueResolved" = true)         THEN     UPDATE "Rental" SET "rentalState" = 'CONCLUDED'::"rentalState" WHERE         "rentalID" = New."issueRentalID";         END IF;     RETURN NEW;     END;     \$\$;</pre>

## Database populated with data

[data.sql](#)

```
--
-- PostgreSQL database dump
--

SET statement_timeout = ;
SET lock_timeout = ;
SET client_encoding = 'UTF8';
SET standard_conforming_strings = on;
SET check_function_bodies = false;
SET client_min_messages = warning;

SET search_path = public, pg_catalog;

--
-- Name: issueCategory; Type: TYPE; Schema: public; Owner: lbaw1653
--

CREATE TYPE "issueCategory" AS ENUM (
    'REFUNDS',
    'COMPLEXISSUES',
    'LASTMINUTECANCELLATION',
    'OTHER'
);
```

```
ALTER TYPE "issueCategory" OWNER TO lbaw1653;

--
-- Name: rentalState; Type: TYPE; Schema: public; Owner: lbaw1653
--

CREATE TYPE "rentalState" AS ENUM (
    'RESERVED',
    'RESERVEDBYMANAGER',
    'CANCELEDBYUSER',
    'CANCELEDBYMANAGER',
    'WAITINGUSERFEEDBACK',
    'SUSPENDEDADMIN',
    'CONCLUDED',
    'CANCELEDBYADMIN'
);

ALTER TYPE "rentalState" OWNER TO lbaw1653;

--
-- Name: surface; Type: TYPE; Schema: public; Owner: lbaw1653
--

CREATE TYPE surface AS ENUM (
    'Synthetic Grass',
    'Grass',
    'Dirt',
    'Floating Wooden Floor',
    'Hardwood',
    'Resilient',
    'Rubber',
    'Linoleum',
    'Vinyl',
    'Other'
);

ALTER TYPE surface OWNER TO lbaw1653;

--
-- Name: surfaceType; Type: TYPE; Schema: public; Owner: lbaw1653
--

CREATE TYPE "surfaceType" AS ENUM (
    'Synthetic Grass',
    'Grass',
    'Dirt',
    'Floating Wooden Floor',
    'Hardwood',
```

```
'Resilient',
'Rubber',
'Linoleum',
'Vinyl',
'Other'
);

ALTER TYPE "surfaceType" OWNER TO lbaw1653;

--
-- Name: banneduser(); Type: FUNCTION; Schema: public; Owner: lbaw1653
--

CREATE FUNCTION banneduser() RETURNS trigger
    LANGUAGE plpgsql
    AS $$BEGIN
IF (OLD."userIsBanned" = false AND New."userIsBanned" = true)
THEN
UPDATE "Rental" SET "rentalState" = 'SUSPENDED BY ADMIN'::"rentalState"
WHERE "rentalUserID" = New."userID";
END IF;
RETURN NEW;
END;
$$;

ALTER FUNCTION public.banneduser() OWNER TO lbaw1653;

--
-- Name: concludedissue(); Type: FUNCTION; Schema: public; Owner:
lbaw1653
--

CREATE FUNCTION concludedissue() RETURNS trigger
    LANGUAGE plpgsql
    AS $$BEGIN
IF(OLD."issueResolved" = false AND New."issueResolved" = true)
THEN
UPDATE "Rental" SET "rentalState" = 'CONCLUDED'::"rentalState" WHERE
"rentalID" = New."issueRentalID";
END IF;
RETURN NEW;
END;
$$;

ALTER FUNCTION public.concludedissue() OWNER TO lbaw1653;

SET default_tablespace = '';
```

```
SET default_with_oids = false;

--
-- Name: Admin; Type: TABLE; Schema: public; Owner: lbaw1653;
-- Tablespace:
--

CREATE TABLE "Admin" (
    "adminUsername" character varying(64) NOT NULL,
    "adminPassword" character varying(256) NOT NULL,
    "adminID" integer NOT NULL,
    "adminAccepted" boolean DEFAULT false NOT NULL
);

ALTER TABLE "Admin" OWNER TO lbaw1653;

--
-- Name: Admin_adminID_seq; Type: SEQUENCE; Schema: public; Owner:
-- lbaw1653
--

CREATE SEQUENCE "Admin_adminID_seq"
    START WITH 1
    INCREMENT BY 1
    NO MINVALUE
    NO MAXVALUE
    CACHE 1;

ALTER TABLE "Admin_adminID_seq" OWNER TO lbaw1653;

--
-- Name: Admin_adminID_seq; Type: SEQUENCE OWNED BY; Schema: public;
-- Owner: lbaw1653
--

ALTER SEQUENCE "Admin_adminID_seq" OWNED BY "Admin"."adminID";

--
-- Name: Equipment; Type: TABLE; Schema: public; Owner: lbaw1653;
-- Tablespace:
--

CREATE TABLE "Equipment" (
    "equipmentName" character varying(64) NOT NULL,
    "equipmentQuantity" smallint NOT NULL,
    "equipmentDetails" character varying(256) DEFAULT NULL::character
    varying,
```



```
"equipmentQuantityUnavailable" smallint DEFAULT NOT NULL,
"equipmentID" bigint NOT NULL,
"equipmentPrice" real NOT NULL,
"equipmentComplexID" integer NOT NULL,
"equipmentInactive" boolean DEFAULT false NOT NULL,
CONSTRAINT "equipmentCheckPrice" CHECK (("equipmentPrice" >
)::real)),
CONSTRAINT "equipmentCheckUnavailable" CHECK
(("equipmentQuantityUnavailable" <= "equipmentQuantity"))
);

ALTER TABLE "Equipment" OWNER TO lbaw1653;

--
-- Name: EquipmentSports; Type: TABLE; Schema: public; Owner: lbaw1653;
--
Tablespace:

CREATE TABLE "EquipmentSports" (
    "equipmentSportsEquipmentID" bigint NOT NULL,
    "equipmentSportsSportID" bigint NOT NULL
);

ALTER TABLE "EquipmentSports" OWNER TO lbaw1653;

--
-- Name: Equipment_equipmentID_seq; Type: SEQUENCE; Schema: public;
--
Owner: lbaw1653

CREATE SEQUENCE "Equipment_equipmentID_seq"
    START WITH 1
    INCREMENT BY 1
    NO MINVALUE
    NO MAXVALUE
    CACHE 1;

ALTER TABLE "Equipment_equipmentID_seq" OWNER TO lbaw1653;

--
-- Name: Equipment_equipmentID_seq; Type: SEQUENCE OWNED BY; Schema:
--
public; Owner: lbaw1653

ALTER SEQUENCE "Equipment_equipmentID_seq" OWNED BY
"Equipment"."equipmentID";
```

```
--
-- Name: Issue; Type: TABLE; Schema: public; Owner: lbaw1653;
Tablespace:
--

CREATE TABLE "Issue" (
    "issueID" bigint NOT NULL,
    "issueSubject" text NOT NULL,
    "issueDescription" text NOT NULL,
    "issueForAdmin" boolean DEFAULT false NOT NULL,
    "issueForManager" boolean DEFAULT false NOT NULL,
    "issueCategory" "issueCategory" NOT NULL,
    "issueRentalID" bigint,
    "issueResolved" boolean DEFAULT false NOT NULL,
    "issueComplexID" bigint,
    "issueUserID" bigint
);

ALTER TABLE "Issue" OWNER TO lbaw1653;

--
-- Name: Issue_issueID_seq; Type: SEQUENCE; Schema: public; Owner:
lbaw1653
--

CREATE SEQUENCE "Issue_issueID_seq"
    START WITH 1
    INCREMENT BY 1
    NO MINVALUE
    NO MAXVALUE
    CACHE 1;

ALTER TABLE "Issue_issueID_seq" OWNER TO lbaw1653;

--
-- Name: Issue_issueID_seq; Type: SEQUENCE OWNED BY; Schema: public;
Owner: lbaw1653
--

ALTER SEQUENCE "Issue_issueID_seq" OWNED BY "Issue"."issueID";

--
-- Name: Manager; Type: TABLE; Schema: public; Owner: lbaw1653;
Tablespace:
--

CREATE TABLE "Manager" (
    "managerComplexID" integer NOT NULL,
```

```
    "managerID" bigint NOT NULL
);

ALTER TABLE "Manager" OWNER TO lbaw1653;

--
-- Name: Municipality; Type: TABLE; Schema: public; Owner: lbaw1653;
-- Tablespace:
--

CREATE TABLE "Municipality" (
    "municipalityName" character varying(64) NOT NULL,
    "municipalityID" integer NOT NULL
);

ALTER TABLE "Municipality" OWNER TO lbaw1653;

--
-- Name: Municipality_municipalityID_seq; Type: SEQUENCE; Schema:
-- public; Owner: lbaw1653
--

CREATE SEQUENCE "Municipality_municipalityID_seq"
    START WITH 1
    INCREMENT BY 1
    NO MINVALUE
    NO MAXVALUE
    CACHE 1;

ALTER TABLE "Municipality_municipalityID_seq" OWNER TO lbaw1653;

--
-- Name: Municipality_municipalityID_seq; Type: SEQUENCE OWNED BY;
-- Schema: public; Owner: lbaw1653
--

ALTER SEQUENCE "Municipality_municipalityID_seq" OWNED BY
"Municipality"."municipalityID";

--
-- Name: Notification; Type: TABLE; Schema: public; Owner: lbaw1653;
-- Tablespace:
--

CREATE TABLE "Notification" (
    "notificationUserID" bigint NOT NULL,
    "notificationText" character varying NOT NULL
```

```
);

ALTER TABLE "Notification" OWNER TO lbaw1653;

--
-- Name: Rental; Type: TABLE; Schema: public; Owner: lbaw1653;
-- Tablespace:
--

CREATE TABLE "Rental" (
    "rentalID" bigint NOT NULL,
    "rentalRating" smallint,
    "rentalStartTime" time without time zone NOT NULL,
    "rentalDuration" interval NOT NULL,
    "rentalDate" date NOT NULL,
    "rentalState" "rentalState" DEFAULT 'RESERVED'::"rentalState" NOT
NULL,
    "rentalSpaceID" bigint NOT NULL,
    "rentalUserID" bigint NOT NULL,
    CONSTRAINT "rentalValidDuration" CHECK (("rentalDuration" >
'00:00:00'::interval)),
    CONSTRAINT "rentalValidRating" CHECK (((("rentalRating" >= 1) AND
("rentalRating" <= 5)))
);

ALTER TABLE "Rental" OWNER TO lbaw1653;

--
-- Name: RentalEquipment; Type: TABLE; Schema: public; Owner: lbaw1653;
-- Tablespace:
--

CREATE TABLE "RentalEquipment" (
    "rentalEquipmentRentalID" bigint NOT NULL,
    "rentalEquipmentEquipmentID" bigint NOT NULL,
    "rentalEquipmentQuantity" smallint NOT NULL,
    CONSTRAINT "rentalEquipmentQuantityPositive" CHECK
(("rentalEquipmentQuantity" > ))
);

ALTER TABLE "RentalEquipment" OWNER TO lbaw1653;

--
-- Name: Rental_rentalID_seq; Type: SEQUENCE; Schema: public; Owner:
lbaw1653
--

CREATE SEQUENCE "Rental_rentalID_seq"
```

```
START WITH 1
INCREMENT BY 1
NO MINVALUE
NO MAXVALUE
CACHE 1;

ALTER TABLE "Rental_rentalID_seq" OWNER TO lbaw1653;

--
-- Name: Rental_rentalID_seq; Type: SEQUENCE OWNED BY; Schema: public;
Owner: lbaw1653
--

ALTER SEQUENCE "Rental_rentalID_seq" OWNED BY "Rental"."rentalID";

--
-- Name: Space; Type: TABLE; Schema: public; Owner: lbaw1653;
Tablespace:
--

CREATE TABLE "Space" (
    "spaceID" bigint NOT NULL,
    "spaceIsCovered" boolean NOT NULL,
    "spaceIsAvailable" boolean NOT NULL,
    "spacePrice" real NOT NULL,
    "spaceComplexID" integer NOT NULL,
    "spaceName" character varying(64),
    "spaceSurfaceType" "surfaceType" NOT NULL,
    CONSTRAINT "spaceCheckPositivePrice" CHECK (("spacePrice" >
)::real))
);

ALTER TABLE "Space" OWNER TO lbaw1653;

--
-- Name: SpaceSports; Type: TABLE; Schema: public; Owner: lbaw1653;
Tablespace:
--

CREATE TABLE "SpaceSports" (
    "spaceSportsSpaceID" bigint NOT NULL,
    "spaceSportsSportID" bigint NOT NULL
);

ALTER TABLE "SpaceSports" OWNER TO lbaw1653;

--
```

```
-- Name: Space_spaceID_seq; Type: SEQUENCE; Schema: public; Owner:
lbaw1653
--

CREATE SEQUENCE "Space_spaceID_seq"
  START WITH 1
  INCREMENT BY 1
  NO MINVALUE
  NO MAXVALUE
  CACHE 1;

ALTER TABLE "Space_spaceID_seq" OWNER TO lbaw1653;

--
-- Name: Space_spaceID_seq; Type: SEQUENCE OWNED BY; Schema: public;
Owner: lbaw1653
--

ALTER SEQUENCE "Space_spaceID_seq" OWNED BY "Space"."spaceID";

--
-- Name: Sport; Type: TABLE; Schema: public; Owner: lbaw1653;
Tablespace:
--

CREATE TABLE "Sport" (
  "sportName" character varying(64) NOT NULL,
  "sportID" integer NOT NULL
);

ALTER TABLE "Sport" OWNER TO lbaw1653;

--
-- Name: Sport_sportID_seq; Type: SEQUENCE; Schema: public; Owner:
lbaw1653
--

CREATE SEQUENCE "Sport_sportID_seq"
  START WITH 1
  INCREMENT BY 1
  NO MINVALUE
  NO MAXVALUE
  CACHE 1;

ALTER TABLE "Sport_sportID_seq" OWNER TO lbaw1653;

--
```

```
-- Name: Sport_sportID_seq; Type: SEQUENCE OWNED BY; Schema: public;
Owner: lbaw1653
--

ALTER SEQUENCE "Sport_sportID_seq" OWNED BY "Sport"."sportID";

--
-- Name: SportsComplex; Type: TABLE; Schema: public; Owner: lbaw1653;
Tablespace:
--

CREATE TABLE "SportsComplex" (
    "complexID" integer NOT NULL,
    "complexName" character varying(64) NOT NULL,
    "complexLocation" character varying(256) NOT NULL,
    "complexOpenOnWeekends" boolean NOT NULL,
    "complexEmail" character varying(64) NOT NULL,
    "complexPhone" character varying(20) NOT NULL,
    "complexPaypal" character varying(64) NOT NULL,
    "complexDescription" character varying(2000) DEFAULT
NULL::character varying,
    "complexOpeningHour" time without time zone NOT NULL,
    "complexClosingHour" time without time zone NOT NULL,
    "complexMunicipalityID" bigint NOT NULL,
    "complexInactive" boolean DEFAULT false NOT NULL,
    CONSTRAINT "complexValidHours" CHECK (("complexClosingHour" >
"complexOpeningHour"))
);

ALTER TABLE "SportsComplex" OWNER TO lbaw1653;

--
-- Name: SportsComplex_complexId_seq; Type: SEQUENCE; Schema: public;
Owner: lbaw1653
--

CREATE SEQUENCE "SportsComplex_complexId_seq"
    START WITH 1
    INCREMENT BY 1
    NO MINVALUE
    NO MAXVALUE
    CACHE 1;

ALTER TABLE "SportsComplex_complexId_seq" OWNER TO lbaw1653;

--
-- Name: SportsComplex_complexId_seq; Type: SEQUENCE OWNED BY; Schema:
public; Owner: lbaw1653
```

```
--  
  
ALTER SEQUENCE "SportsComplex_complexId_seq" OWNED BY  
"SportsComplex"."complexID";  
  
--  
-- Name: User; Type: TABLE; Schema: public; Owner: lbaw1653;  
-- Tablespace:  
--  
  
CREATE TABLE "User" (  
    "userUsername" character varying(64) NOT NULL,  
    "userPassword" character varying(256) NOT NULL,  
    "userName" character varying(64) NOT NULL,  
    "userEmail" character varying(64) NOT NULL,  
    "userPhone" character varying(20) DEFAULT NULL::character varying,  
    "userIsBanned" boolean DEFAULT false NOT NULL,  
    "userID" integer NOT NULL,  
    "userMunicipalityID" bigint  
);  
  
ALTER TABLE "User" OWNER TO lbaw1653;  
  
--  
-- Name: User_userID_seq; Type: SEQUENCE; Schema: public; Owner:  
-- lbaw1653  
--  
  
CREATE SEQUENCE "User_userID_seq"  
    START WITH 1  
    INCREMENT BY 1  
    NO MINVALUE  
    NO MAXVALUE  
    CACHE 1;  
  
ALTER TABLE "User_userID_seq" OWNER TO lbaw1653;  
  
--  
-- Name: User_userID_seq; Type: SEQUENCE OWNED BY; Schema: public;  
-- Owner: lbaw1653  
--  
  
ALTER SEQUENCE "User_userID_seq" OWNED BY "User"."userID";  
  
--  
-- Name: adminID; Type: DEFAULT; Schema: public; Owner: lbaw1653  
--
```



```
ALTER TABLE ONLY "Admin" ALTER COLUMN "adminID" SET DEFAULT
nextval('"Admin_adminID_seq"'::regclass);

--
-- Name: equipmentID; Type: DEFAULT; Schema: public; Owner: lbaw1653
--

ALTER TABLE ONLY "Equipment" ALTER COLUMN "equipmentID" SET DEFAULT
nextval('"Equipment_equipmentID_seq"'::regclass);

--
-- Name: issueID; Type: DEFAULT; Schema: public; Owner: lbaw1653
--

ALTER TABLE ONLY "Issue" ALTER COLUMN "issueID" SET DEFAULT
nextval('"Issue_issueID_seq"'::regclass);

--
-- Name: municipalityID; Type: DEFAULT; Schema: public; Owner: lbaw1653
--

ALTER TABLE ONLY "Municipality" ALTER COLUMN "municipalityID" SET
DEFAULT nextval('"Municipality_municipalityID_seq"'::regclass);

--
-- Name: rentalID; Type: DEFAULT; Schema: public; Owner: lbaw1653
--

ALTER TABLE ONLY "Rental" ALTER COLUMN "rentalID" SET DEFAULT
nextval('"Rental_rentalID_seq"'::regclass);

--
-- Name: spaceID; Type: DEFAULT; Schema: public; Owner: lbaw1653
--

ALTER TABLE ONLY "Space" ALTER COLUMN "spaceID" SET DEFAULT
nextval('"Space_spaceID_seq"'::regclass);

--
-- Name: sportID; Type: DEFAULT; Schema: public; Owner: lbaw1653
--

ALTER TABLE ONLY "Sport" ALTER COLUMN "sportID" SET DEFAULT
nextval('"Sport_sportID_seq"'::regclass);
```

```
--
-- Name: complexID; Type: DEFAULT; Schema: public; Owner: lbaw1653
--

ALTER TABLE ONLY "SportsComplex" ALTER COLUMN "complexID" SET DEFAULT
nextval('"SportsComplex_complexId_seq"'::regclass);

--
-- Name: userID; Type: DEFAULT; Schema: public; Owner: lbaw1653
--

ALTER TABLE ONLY "User" ALTER COLUMN "userID" SET DEFAULT
nextval('"User_userID_seq"'::regclass);

--
-- Data for Name: Admin; Type: TABLE DATA; Schema: public; Owner:
lbaw1653
--

INSERT INTO "Admin" VALUES ('adam',
'$2y$10$8lej dILJHAADNKgmN3j0LusGRJFBwDeGh2rlpXqDnD.wmFAQi/lsu', 6,
true);
INSERT INTO "Admin" VALUES ('EduardoAdmin',
'$2y$10$R2kiKXVVICuNf7VKRYjLseSTKm7dJV5ZDnT88EZnqNFJNMn28uVVi', 8,
true);
INSERT INTO "Admin" VALUES ('newAdmin',
'$2y$10$zyztT/5zJtiFGQQ5IE4L2uQITSRZblpXsPxPjMT6pkenLHJtwiz70', 9,
false);

--
-- Name: Admin_adminID_seq; Type: SEQUENCE SET; Schema: public; Owner:
lbaw1653
--

SELECT pg_catalog.setval('"Admin_adminID_seq"', 9, true);

--
-- Data for Name: Equipment; Type: TABLE DATA; Schema: public; Owner:
lbaw1653
--

INSERT INTO "Equipment" VALUES ('Bicicleta', 30, 'Bicicletas para
prática de desporto no complexo desportivo.', , 79, 25, 99, false);
INSERT INTO "Equipment" VALUES ('Tenis ball', 20, '', , 80, 2, 100,
false);
```

```

INSERT INTO "Equipment" VALUES ('Tennis racket', 15, '', , 81, 5, 100, false);
INSERT INTO "Equipment" VALUES ('Basketball ball', 50, '', , 82, 5, 100, false);
INSERT INTO "Equipment" VALUES ('Football ball', 25, '', , 83, 5, 100, false);
INSERT INTO "Equipment" VALUES ('Training vest', 100, '', , 84, 0.5, 100, false);
INSERT INTO "Equipment" VALUES ('Volleyball ball', 20, '', , 85, 2, 100, false);
INSERT INTO "Equipment" VALUES ('Volleyball net', 2, '', , 86, 1, 100, false);
INSERT INTO "Equipment" VALUES ('Equipamento', 10, 'Detalhes', , 87, 5, 104, false);
INSERT INTO "Equipment" VALUES ('Bola De Basket', 5, '', , 88, 2, 105, false);
INSERT INTO "Equipment" VALUES ('Colete', 10, '', , 89, 1, 105, false);
INSERT INTO "Equipment" VALUES ('Bolas de basketball', 5, 'Bola de basketball', , 90, 4, 106, false);
INSERT INTO "Equipment" VALUES ('Colete', 10, 'Colete para jogos', , 91, 2, 106, false);
INSERT INTO "Equipment" VALUES ('Raquete', 6, 'Raquete de Tennis Simples', , 92, 5, 107, false);
INSERT INTO "Equipment" VALUES ('Bola de Tennis', 3, 'Bola de Tennis Simples', , 93, 2, 107, false);

--
-- Data for Name: EquipmentSports; Type: TABLE DATA; Schema: public; Owner: lbaw1653
--

INSERT INTO "EquipmentSports" VALUES (79, 18);
INSERT INTO "EquipmentSports" VALUES (80, 14);
INSERT INTO "EquipmentSports" VALUES (81, 14);
INSERT INTO "EquipmentSports" VALUES (82, 3);
INSERT INTO "EquipmentSports" VALUES (83, 7);
INSERT INTO "EquipmentSports" VALUES (85, 2);
INSERT INTO "EquipmentSports" VALUES (86, 2);
INSERT INTO "EquipmentSports" VALUES (87, 1);
INSERT INTO "EquipmentSports" VALUES (87, 2);
INSERT INTO "EquipmentSports" VALUES (87, 3);
INSERT INTO "EquipmentSports" VALUES (87, 4);
INSERT INTO "EquipmentSports" VALUES (87, 5);
INSERT INTO "EquipmentSports" VALUES (87, 6);
INSERT INTO "EquipmentSports" VALUES (87, 7);
INSERT INTO "EquipmentSports" VALUES (87, 8);
INSERT INTO "EquipmentSports" VALUES (87, 9);
INSERT INTO "EquipmentSports" VALUES (87, 10);
INSERT INTO "EquipmentSports" VALUES (87, 11);
INSERT INTO "EquipmentSports" VALUES (87, 12);

```

```
INSERT INTO "EquipmentSports" VALUES (87, 13);
INSERT INTO "EquipmentSports" VALUES (87, 14);
INSERT INTO "EquipmentSports" VALUES (87, 18);
INSERT INTO "EquipmentSports" VALUES (87, 19);
INSERT INTO "EquipmentSports" VALUES (87, 20);
INSERT INTO "EquipmentSports" VALUES (87, 21);
INSERT INTO "EquipmentSports" VALUES (88, 3);
INSERT INTO "EquipmentSports" VALUES (89, 3);
INSERT INTO "EquipmentSports" VALUES (90, 3);
INSERT INTO "EquipmentSports" VALUES (91, 3);
INSERT INTO "EquipmentSports" VALUES (92, 14);
INSERT INTO "EquipmentSports" VALUES (93, 14);

--
-- Name: Equipment_equipementID_seq; Type: SEQUENCE SET; Schema: public;
Owner: lbaw1653
--

SELECT pg_catalog.setval('"Equipment_equipementID_seq"', 93, true);

--
-- Data for Name: Issue; Type: TABLE DATA; Schema: public; Owner:
lbaw1653
--

INSERT INTO "Issue" VALUES (46, 'Unclean bathroom', 'The service on the
pool was amazing but you should clean bathrooms more often :/', false,
true, 'COMPLEXISSUES', 78, false, NULL, NULL);
INSERT INTO "Issue" VALUES (37, 'Bola por arranjar', 'A bola amarela
está por arranjar!', false, true, 'COMPLEXISSUES', NULL, true, 105,
NULL);
INSERT INTO "Issue" VALUES (40, 'Money Back', 'I want my money back!',
true, true, 'REFUNDS', 74, false, NULL, NULL);
INSERT INTO "Issue" VALUES (41, 'Issue', 'Issue', false, true,
'COMPLEXISSUES', NULL, false, 99, NULL);
INSERT INTO "Issue" VALUES (42, 'Refund', 'The service was awfull, i
want a refund', true, false, 'REFUNDS', NULL, false, NULL, 94);

--
-- Name: Issue_issueID_seq; Type: SEQUENCE SET; Schema: public; Owner:
lbaw1653
--

SELECT pg_catalog.setval('"Issue_issueID_seq"', 46, true);

--
-- Data for Name: Manager; Type: TABLE DATA; Schema: public; Owner:
```

```
lbaw1653
```

```
--
```

```
INSERT INTO "Manager" VALUES (99, 94);
INSERT INTO "Manager" VALUES (100, 94);
INSERT INTO "Manager" VALUES (101, 95);
INSERT INTO "Manager" VALUES (102, 95);
INSERT INTO "Manager" VALUES (103, 96);
INSERT INTO "Manager" VALUES (104, 93);
INSERT INTO "Manager" VALUES (105, 101);
INSERT INTO "Manager" VALUES (106, 103);
INSERT INTO "Manager" VALUES (107, 106);
```

```
--
```

```
-- Data for Name: Municipality; Type: TABLE DATA; Schema: public;
Owner: lbaw1653
```

```
--
```

```
INSERT INTO "Municipality" VALUES ('Aveiro', 12);
INSERT INTO "Municipality" VALUES ('Beja', 14);
INSERT INTO "Municipality" VALUES ('Braga', 15);
INSERT INTO "Municipality" VALUES ('Bragança', 16);
INSERT INTO "Municipality" VALUES ('Castelo Branco', 17);
INSERT INTO "Municipality" VALUES ('Coimbra', 18);
INSERT INTO "Municipality" VALUES ('Évora', 19);
INSERT INTO "Municipality" VALUES ('Faro', 20);
INSERT INTO "Municipality" VALUES ('Guarda', 21);
INSERT INTO "Municipality" VALUES ('Leiria', 22);
INSERT INTO "Municipality" VALUES ('Lisboa', 23);
INSERT INTO "Municipality" VALUES ('Portalegre', 24);
INSERT INTO "Municipality" VALUES ('Porto', 25);
INSERT INTO "Municipality" VALUES ('Santarém', 26);
INSERT INTO "Municipality" VALUES ('Setúbal', 27);
INSERT INTO "Municipality" VALUES ('Viana do Castelo', 28);
INSERT INTO "Municipality" VALUES ('Vila Real', 29);
INSERT INTO "Municipality" VALUES ('Viseu', 30);
```

```
--
```

```
-- Name: Municipality_municipalityID_seq; Type: SEQUENCE SET; Schema:
public; Owner: lbaw1653
```

```
--
```

```
SELECT pg_catalog.setval('"Municipality_municipalityID_seq"', 30,
true);
```

```
--
```

```
-- Data for Name: Notification; Type: TABLE DATA; Schema: public;
Owner: lbaw1653
```

```
--

INSERT INTO "Notification" VALUES (8, 'You recently received a new
issue with the subject: cdv.');
```

```
INSERT INTO "Notification" VALUES (9, 'You recently received a new
issue with the subject: cdv.');
```

```
--

-- Data for Name: Rental; Type: TABLE DATA; Schema: public; Owner:
lbaw1653
--

INSERT INTO "Rental" VALUES (74, 2, '08:00:00', '12:00:00',
'2020-01-20', 'CONCLUDED', 123, 102);
INSERT INTO "Rental" VALUES (75, 3, '14:00:00', '00:02:45',
'2020-11-12', 'CANCELEDBYADMIN', 123, 93);
INSERT INTO "Rental" VALUES (78, 4, '10:00:00', '01:00:00',
'2017-06-02', 'CONCLUDED', 131, 94);
INSERT INTO "Rental" VALUES (77, NULL, '15:00:00', '01:30:00',
'2017-07-03', 'RESERVED', 138, 94);
INSERT INTO "Rental" VALUES (76, NULL, '16:00:00', '00:30:00',
'2017-07-30', 'CANCELEDBYUSER', 124, 94);
INSERT INTO "Rental" VALUES (79, NULL, '16:00:00', '00:20:00',
'2017-08-20', 'RESERVED', 127, 94);

--

-- Data for Name: RentalEquipment; Type: TABLE DATA; Schema: public;
Owner: lbaw1653
--

INSERT INTO "RentalEquipment" VALUES (74, 88, 1);
INSERT INTO "RentalEquipment" VALUES (74, 89, 5);
INSERT INTO "RentalEquipment" VALUES (76, 79, 3);
INSERT INTO "RentalEquipment" VALUES (77, 90, 4);
INSERT INTO "RentalEquipment" VALUES (77, 91, 6);
INSERT INTO "RentalEquipment" VALUES (79, 85, 2);

--

-- Name: Rental_rentalID_seq; Type: SEQUENCE SET; Schema: public;
Owner: lbaw1653
--

SELECT pg_catalog.setval('Rental_rentalID_seq', 79, true);

--

-- Data for Name: Space; Type: TABLE DATA; Schema: public; Owner:
lbaw1653
```

```
--
INSERT INTO "Space" VALUES (123, true, true, 5, 105, 'BobSpace1',
'Dirt');
INSERT INTO "Space" VALUES (125, true, true, 30, 99, 'Piscina CAR',
'Other');
INSERT INTO "Space" VALUES (126, true, true, 5, 104, 'Espaço do
Francisco', 'Hardwood');
INSERT INTO "Space" VALUES (137, true, true, 25, 106, 'PeterSpace1',
'Hardwood');
INSERT INTO "Space" VALUES (138, false, true, 15, 106, 'PeterSpace2',
'Other');
INSERT INTO "Space" VALUES (131, true, true, 10, 102, 'Piscinas
Municipais', 'Other');
INSERT INTO "Space" VALUES (132, false, true, 20, 102, 'Courts de
Ténis', 'Synthetic Grass');
INSERT INTO "Space" VALUES (128, true, true, 15, 100, 'Pavilhão da
Pampilhosa', 'Vinyl');
INSERT INTO "Space" VALUES (127, true, true, 10, 100, 'Pavilhão do
Luso', 'Floating Wooden Floor');
INSERT INTO "Space" VALUES (129, false, false, 20, 100, 'Ténis Jorge
Humberto', 'Other');
INSERT INTO "Space" VALUES (130, false, true, 20, 101, 'Campo Futebol
do Complexo Desportivo da Tocha', 'Synthetic Grass');
INSERT INTO "Space" VALUES (134, false, true, 25, 102, 'Campo de
Futebol de 11', 'Synthetic Grass');
INSERT INTO "Space" VALUES (136, true, true, 45, 102, 'Pavilhão
Municipal de Desportos', 'Vinyl');
INSERT INTO "Space" VALUES (135, false, true, 70, 102, 'Estádio
Municipal de Anadia', 'Synthetic Grass');
INSERT INTO "Space" VALUES (133, true, true, 15, 102, 'Campo de Futebol
de 7', 'Floating Wooden Floor');
INSERT INTO "Space" VALUES (124, true, true, 70, 99, 'Velódromo',
'Floating Wooden Floor');
INSERT INTO "Space" VALUES (139, true, true, 15, 107, 'Court 1',
'Grass');
INSERT INTO "Space" VALUES (140, false, false, 15, 107, 'Court 2',
'Grass');

--
-- Data for Name: SpaceSports; Type: TABLE DATA; Schema: public; Owner:
lbaw1653
--

INSERT INTO "SpaceSports" VALUES (123, 3);
INSERT INTO "SpaceSports" VALUES (124, 11);
INSERT INTO "SpaceSports" VALUES (124, 18);
INSERT INTO "SpaceSports" VALUES (125, 19);
INSERT INTO "SpaceSports" VALUES (126, 11);
INSERT INTO "SpaceSports" VALUES (127, 2);
```

```
INSERT INTO "SpaceSports" VALUES (128, 1);
INSERT INTO "SpaceSports" VALUES (128, 2);
INSERT INTO "SpaceSports" VALUES (128, 3);
INSERT INTO "SpaceSports" VALUES (128, 7);
INSERT INTO "SpaceSports" VALUES (127, 3);
INSERT INTO "SpaceSports" VALUES (127, 7);
INSERT INTO "SpaceSports" VALUES (129, 14);
INSERT INTO "SpaceSports" VALUES (130, 7);
INSERT INTO "SpaceSports" VALUES (130, 8);
INSERT INTO "SpaceSports" VALUES (130, 9);
INSERT INTO "SpaceSports" VALUES (131, 19);
INSERT INTO "SpaceSports" VALUES (132, 14);
INSERT INTO "SpaceSports" VALUES (133, 20);
INSERT INTO "SpaceSports" VALUES (134, 7);
INSERT INTO "SpaceSports" VALUES (135, 7);
INSERT INTO "SpaceSports" VALUES (135, 8);
INSERT INTO "SpaceSports" VALUES (135, 9);
INSERT INTO "SpaceSports" VALUES (136, 1);
INSERT INTO "SpaceSports" VALUES (136, 2);
INSERT INTO "SpaceSports" VALUES (136, 3);
INSERT INTO "SpaceSports" VALUES (136, 12);
INSERT INTO "SpaceSports" VALUES (136, 20);
INSERT INTO "SpaceSports" VALUES (137, 3);
INSERT INTO "SpaceSports" VALUES (138, 3);
INSERT INTO "SpaceSports" VALUES (139, 14);
INSERT INTO "SpaceSports" VALUES (140, 14);

--
-- Name: Space_spaceID_seq; Type: SEQUENCE SET; Schema: public; Owner:
lbaw1653
--

SELECT pg_catalog.setval('"Space_spaceID_seq"', 140, true);

--
-- Data for Name: Sport; Type: TABLE DATA; Schema: public; Owner:
lbaw1653
--

INSERT INTO "Sport" VALUES ('Badminton', 1);
INSERT INTO "Sport" VALUES ('Volleyball', 2);
INSERT INTO "Sport" VALUES ('Basketball', 3);
INSERT INTO "Sport" VALUES ('Baseball', 4);
INSERT INTO "Sport" VALUES ('Boxing', 5);
INSERT INTO "Sport" VALUES ('Football', 7);
INSERT INTO "Sport" VALUES ('American football', 8);
INSERT INTO "Sport" VALUES ('Rugby football', 9);
INSERT INTO "Sport" VALUES ('Golf', 10);
INSERT INTO "Sport" VALUES ('Gymnastics', 11);
```



```

INSERT INTO "Sport" VALUES ('Handball', 12);
INSERT INTO "Sport" VALUES ('Hockey', 13);
INSERT INTO "Sport" VALUES ('Cycling', 18);
INSERT INTO "Sport" VALUES ('Swimming', 19);
INSERT INTO "Sport" VALUES ('Indoor Football', 20);
INSERT INTO "Sport" VALUES ('Karting', 21);
INSERT INTO "Sport" VALUES ('Fencing', 6);
INSERT INTO "Sport" VALUES ('Tennis', 14);

--
-- Name: Sport_sportID_seq; Type: SEQUENCE SET; Schema: public; Owner:
lbaw1653
--

SELECT pg_catalog.setval('"Sport_sportID_seq"', 21, true);

--
-- Data for Name: SportsComplex; Type: TABLE DATA; Schema: public;
Owner: lbaw1653
--

INSERT INTO "SportsComplex" VALUES (101, 'Complexo Desportivo da
Tocha', 'Zona Industrial da Tocha', true, 'tocha@gmail.com',
'231410100', 'tocha@gmail.com', 'O Complexo Desportivo da Tocha é um
dos melhor dimensionados e equipados parques desportivos da Região
Centro.
É constituído por um campo em relva natural de dimensões oficiais para
a realização de competições internacionais, com uma bancada central com
capacidade para 1.000 espectadores, e um campo secundário em relva
artificial, também com dimensões oficiais para a realização de
competições internacionais em futebol, rãguebi e outros desportos. ',
'12:00:00', '20:00:00', 18, false);
INSERT INTO "SportsComplex" VALUES (102, 'Complexo Desportivo de
Anadia', '3780-243 Anadia', true, 'cmadesporto@mail.telepac.pt',
'231503935', 'cmadesporto@mail.telepac.pt', 'Anadia, Município do
Desporto - O Complexo Desportivo de Anadia tem sido uma mais valia na
promoção e prática desportiva. É um espaço dinâmico que recebe equipas
nacionais e internacionais entre outras, nas modalidades de futebol e
basquetebol nas várias categorias femininas e masculinas.', '09:00:00',
'21:30:00', 12, false);
INSERT INTO "SportsComplex" VALUES (103, 'Parque Expo-desportivo de S.
Mateus', 'Rua dos Bombeiros Voluntarios, Cantanhede', false,
'saomateus@gmail.com', '231410100', 'saomateus@gmail.com', 'Espaço
dedicado à realização de actividades desportivas, nomeadamente futebol,
bem como à realização de eventos de exposição/feiras. Possui campo
relvado, campo pelado e pista de modelismo.', '09:00:00', '18:00:00',
18, false);
INSERT INTO "SportsComplex" VALUES (100, 'Centro de Estágios do Luso',
'Valinho - Luso', true, 'estagiosluso@gmail.com', '231937950',

```

'estagiosluso@gmail.com', 'O Centro de Treinos/Estágios do Luso construído por ocasião do Euro 2004 foi executado de acordo com as normas e regulamentos da UEFA, tendo ido considerado Centro Oficial de treinos.

Constituído por dois campos de relva natural e pista de tartan, apoiados por modernos balneários, sala de imprensa e de reuniões, bar e sala de musculação, conta ainda com auditório para 90 pessoas e ponto wireless com acesso á Internet. ', '09:00:00', '22:00:00', 12, false);

INSERT INTO "SportsComplex" VALUES (104, 'Complexo do Francisco', 'Rua Santa Luzia, 503, Penafiel', false, 'franciscotomequeiros@gmail.com', '919705360', 'grifferz2@gmail.com', 'Uma casa de desporto', '12:00:00', '20:00:00', 25, false);

INSERT INTO "SportsComplex" VALUES (99, 'CAR - Velódromo Nacional', 'Rua Ivo Neves, nº 405 - Sangalhos', false, 'velodromonacional@gmail.com', '234738218', 'velodromonacional@gmail.com', 'Este novo espaço, é Centro de Alto Rendimento das modalidades de Ciclismo, Esgrima, Judo, Ginástica, Trampolins e Desportos Acrobáticos. Para além da pista coberta, com 250m, para a modalidade de ciclismo, dispõe-se de uma área polivalente com 1100m2.', '09:00:00', '17:30:00', 12, false);

INSERT INTO "SportsComplex" VALUES (105, 'Complexo do Bob', 'Rua do Complexo do Bob', true, 'bobcomplex@gmail.com', '913034514', 'bobcomplex@gmail.com', 'O melhor complexo desportivo de Faro', '08:00:00', '20:00:00', 20, false);

INSERT INTO "SportsComplex" VALUES (106, 'Complexo do Peter', 'Rua Nuno Alvares, 34, Gondomar', true, 'petercomplex@live.com.pt', '936528547', 'ademarpereira@live.com.pt', 'Complexo desportivo para pequenos eventos.', '08:00:00', '22:00:00', 25, false);

INSERT INTO "SportsComplex" VALUES (107, 'Tennis do Abilio', 'Rua Abilio', true, 'abilio@gmail.com', '913031245', 'abilio@gmail.com', 'Campo de tennis muito bom!', '08:00:00', '20:00:00', 21, false);

--

-- Name: SportsComplex\_complexId\_seq; Type: SEQUENCE SET; Schema: public; Owner: lbaw1653

--

SELECT pg\_catalog.setval('SportsComplex\_complexId\_seq', 107, true);

--

-- Data for Name: User; Type: TABLE DATA; Schema: public; Owner: lbaw1653

--

INSERT INTO "User" VALUES ('xjupiter', '\$2y\$10\$AW4DmjABweDL4q51GSsYQ.1x.EHExQ4gdztGaaF9Q0sQBR0sPIqMu', 'Eduardo Leite', 'embleite@gmail.com', '914457886', false, 95, 25);

INSERT INTO "User" VALUES ('maria123456', '\$2y\$10\$Slq6qrJfHE8ZcmychCGa8e9onCzUWmuwbAyyjR0.wyKjxSts7DPPbC', 'Maria

```

Costa', 'maria@gmail.com', '935687459', false, 96, 30);
INSERT INTO "User" VALUES ('mael23',
'$2y$10$fm2mN4v9A.Q1tAHwK0MG7eInrrkCTf077Cp7Rm/B1dMcKZHKrLCLk',
'Cristina', 'cagfsag@hotmail.com', '938592400', false, 97, 26);
INSERT INTO "User" VALUES ('andreia13',
'$2y$10$zccMpSdblsaNRr51Ja/P20ET/3FSu4aU95mxwP0UVUgR8ZI8ZDY02',
'Andreia Rodrigues', 'andreiacarodrigues@gmail.com', '912255661',
false, 94, 12);
INSERT INTO "User" VALUES ('ademar1',
'$2y$10$Xwes0QgHjKfSs0zAtDurnuXc./EQ1v/PfBdyUdxehXpGJv3Zw1VpS',
'Ademar', 'ademarjoapereira@gmail.com', '934088355', false, 98, 25);
INSERT INTO "User" VALUES ('Francisco',
'$2y$10$ZcqK7yq2.YTFxZQ2h03.U0GmorbS8VmjhZ2SDwnX0VKLUo2HPM/tK',
'Francisco Queirós', 'grifferz2@gmail.com', '919705360', false, 93,
25);
INSERT INTO "User" VALUES ('Tr1b0lt',
'$2y$10$IZDwbCk5yBMZk2gv480Kmei63UG9tdpgi3RaRKWRM9lDeQ1uK4kCi', 'João
Maria', 'Tr1b0lt@gmail.com', '913034512', false, 99, 23);
INSERT INTO "User" VALUES ('admin',
'$2y$10$MhbJ44VGtxKXGSjCwe7Gv06Q2EvCwC8tEBerZ/Lrz8o63EWyKnxeS', 'a',
'a@gmail.com', '111111111', false, 100, 28);
INSERT INTO "User" VALUES ('DaveBuyer',
'$2y$10$Ag8qTRLGA.0AIFQhYnP8Aultgfvzk9g6eAb0zmCiTL5q2CLYEBYT.', 'Dave',
'DaveBuyer@gmail.com', '913034213', false, 102, 20);
INSERT INTO "User" VALUES ('PeterOwner',
'$2y$10$MyRrgfpoXgLtnQGEi0Pzle0EjJ7tYWGS.cRQBZBlSLriaP9ChjW6i',
'Peter', 'ademapereira@live.com.pt', '934088354', false, 103, 25);
INSERT INTO "User" VALUES ('AdemarBuyer',
'$2y$10$/YIZaAN5fx0TEs0Dx2hnh0MDYeCPDKKjqF70AEYQjYYntb/j/p8Dy',
'Ademar', 'ademarleal@hotmail.com', '936985698', false, 104, 25);
INSERT INTO "User" VALUES ('JoaoBuyer',
'$2y$10$lkW14REQEp5SRvolpFsYX.vcg79QjkqzFr0zEI8840kZ1pYGVz//i', 'João',
'joapereira@gmail.com', '936985478', false, 105, 25);
INSERT INTO "User" VALUES ('BobOwner',
'$2y$10$Xxz4mXaNw56L3JZpm9d0o.SMdCV1aRIvcA8VbBeuszPobK1WF4n1a', 'Bob',
'BobOwner@gmail.com', '913011111', false, 101, 20);
INSERT INTO "User" VALUES ('Abil',
'$2y$10$At/F5rHnJLqmhSlxC/10BeGq5ZVYEqt5Lm1tKyxnYHyyhznExXX8u',
'Abilio', 'abilio@gmail.com', '913036543', false, 106, 21);

--
-- Name: User_userID_seq; Type: SEQUENCE SET; Schema: public; Owner:
lbaw1653
--

SELECT pg_catalog.setval('"User_userID_seq"', 106, true);

--
-- Name: Admin_pkey; Type: CONSTRAINT; Schema: public; Owner: lbaw1653;

```

Tablespace:

--

```
ALTER TABLE ONLY "Admin"  
    ADD CONSTRAINT "Admin_pkey" PRIMARY KEY ("adminID");
```

--

-- Name: EquipmentSports\_pkey; Type: CONSTRAINT; Schema: public; Owner:  
lbaw1653; Tablespace:

--

```
ALTER TABLE ONLY "EquipmentSports"  
    ADD CONSTRAINT "EquipmentSports_pkey" PRIMARY KEY  
("equipmentSportsEquipmentID", "equipmentSportsSportID");
```

--

-- Name: Equipment\_pkey; Type: CONSTRAINT; Schema: public; Owner:  
lbaw1653; Tablespace:

--

```
ALTER TABLE ONLY "Equipment"  
    ADD CONSTRAINT "Equipment_pkey" PRIMARY KEY ("equipmentID");
```

--

-- Name: Issue\_pkey; Type: CONSTRAINT; Schema: public; Owner: lbaw1653;  
Tablespace:

--

```
ALTER TABLE ONLY "Issue"  
    ADD CONSTRAINT "Issue_pkey" PRIMARY KEY ("issueID");
```

--

-- Name: Manager\_pkey; Type: CONSTRAINT; Schema: public; Owner:  
lbaw1653; Tablespace:

--

```
ALTER TABLE ONLY "Manager"  
    ADD CONSTRAINT "Manager_pkey" PRIMARY KEY ("managerComplexID",  
"managerID");
```

--

-- Name: Municipality\_pkey; Type: CONSTRAINT; Schema: public; Owner:  
lbaw1653; Tablespace:

--

```
ALTER TABLE ONLY "Municipality"
```

```
    ADD CONSTRAINT "Municipality_pkey" PRIMARY KEY ("municipalityID");

--
-- Name: Notification_pkey; Type: CONSTRAINT; Schema: public; Owner:
lbaw1653; Tablespace:
--

ALTER TABLE ONLY "Notification"
    ADD CONSTRAINT "Notification_pkey" PRIMARY KEY
("notificationUserID", "notificationText");

--
-- Name: RentalEquipment_pkey; Type: CONSTRAINT; Schema: public; Owner:
lbaw1653; Tablespace:
--

ALTER TABLE ONLY "RentalEquipment"
    ADD CONSTRAINT "RentalEquipment_pkey" PRIMARY KEY
("rentalEquipmentRentalID", "rentalEquipmentEquipmentID");

--
-- Name: Rental_pkey; Type: CONSTRAINT; Schema: public; Owner:
lbaw1653; Tablespace:
--

ALTER TABLE ONLY "Rental"
    ADD CONSTRAINT "Rental_pkey" PRIMARY KEY ("rentalID");

--
-- Name: SpaceSports_pkey; Type: CONSTRAINT; Schema: public; Owner:
lbaw1653; Tablespace:
--

ALTER TABLE ONLY "SpaceSports"
    ADD CONSTRAINT "SpaceSports_pkey" PRIMARY KEY
("spaceSportsSpaceID", "spaceSportsSportID");

--
-- Name: Space_pkey; Type: CONSTRAINT; Schema: public; Owner: lbaw1653;
Tablespace:
--

ALTER TABLE ONLY "Space"
    ADD CONSTRAINT "Space_pkey" PRIMARY KEY ("spaceID");
```

```
--
-- Name: Sport_pkey; Type: CONSTRAINT; Schema: public; Owner: lbaw1653;
Tablespace:
--

ALTER TABLE ONLY "Sport"
    ADD CONSTRAINT "Sport_pkey" PRIMARY KEY ("sportID");

--
-- Name: SportsComplex_pkey; Type: CONSTRAINT; Schema: public; Owner:
lbaw1653; Tablespace:
--

ALTER TABLE ONLY "SportsComplex"
    ADD CONSTRAINT "SportsComplex_pkey" PRIMARY KEY ("complexID");

--
-- Name: User_pkey; Type: CONSTRAINT; Schema: public; Owner: lbaw1653;
Tablespace:
--

ALTER TABLE ONLY "User"
    ADD CONSTRAINT "User_pkey" PRIMARY KEY ("userID");

--
-- Name: User_userEmail_key; Type: CONSTRAINT; Schema: public; Owner:
lbaw1653; Tablespace:
--

ALTER TABLE ONLY "User"
    ADD CONSTRAINT "User_userEmail_key" UNIQUE ("userEmail");

--
-- Name: User_userUsername_key; Type: CONSTRAINT; Schema: public;
Owner: lbaw1653; Tablespace:
--

ALTER TABLE ONLY "User"
    ADD CONSTRAINT "User_userUsername_key" UNIQUE ("userUsername");

--
-- Name: equipmentInfoIndex; Type: INDEX; Schema: public; Owner:
lbaw1653; Tablespace:
--

CREATE INDEX "equipmentInfoIndex" ON "Equipment" USING btree
```

```
("equipmentComplexID", "equipmentID", "equipmentQuantity",
"equipmentQuantityUnavailable");

--
-- Name: issueRentalIndex; Type: INDEX; Schema: public; Owner:
lbaw1653; Tablespace:
--

CREATE INDEX "issueRentalIndex" ON "Issue" USING hash
("issueRentalID");

--
-- Name: rentalSpaceTimeIndex; Type: INDEX; Schema: public; Owner:
lbaw1653; Tablespace:
--

CREATE INDEX "rentalSpaceTimeIndex" ON "Rental" USING btree
("rentalID", "rentalSpaceID", "rentalStartTime", "rentalDuration");

--
-- Name: rentalUserIndex; Type: INDEX; Schema: public; Owner: lbaw1653;
Tablespace:
--

CREATE INDEX "rentalUserIndex" ON "Rental" USING hash ("rentalUserID");

--
-- Name: sportComplexSearchIndex; Type: INDEX; Schema: public; Owner:
lbaw1653; Tablespace:
--

CREATE INDEX "sportComplexSearchIndex" ON "SportsComplex" USING hash
("complexLocation");

--
-- Name: sportcomplexsearchindexfulltext; Type: INDEX; Schema: public;
Owner: lbaw1653; Tablespace:
--

CREATE INDEX sportcomplexsearchindexfulltext ON "SportsComplex" USING
gin (to_tsvector('english'::regconfig, (((("complexName")::text || '
'::text) || ("complexLocation")::text) || ' '::text) ||
("complexDescription")::text)));

--
```

```
-- Name: sportcomplexsearchindexfulltextweighted; Type: INDEX; Schema:
public; Owner: lbaw1653; Tablespace:
--

CREATE INDEX sportcomplexsearchindexfulltextweighted ON "SportsComplex"
USING gin (((setweight(to_tsvector('english'::regconfig,
("complexName")::text), 'A'::"char") ||
setweight(to_tsvector('english'::regconfig, ("complexLocation")::text),
'B'::"char")) || setweight(to_tsvector('english'::regconfig,
("complexDescription")::text), 'C'::"char"))));

--
-- Name: sportcomplexsearchindexname; Type: INDEX; Schema: public;
Owner: lbaw1653; Tablespace:
--

CREATE INDEX sportcomplexsearchindexname ON "SportsComplex" USING gin
(to_tsvector('english'::regconfig, ("complexName")::text));

--
-- Name: banneduser; Type: TRIGGER; Schema: public; Owner: lbaw1653
--

CREATE TRIGGER banneduser AFTER UPDATE OF "userIsBanned" ON "User" FOR
EACH ROW EXECUTE PROCEDURE banneduser();

--
-- Name: concludedissue; Type: TRIGGER; Schema: public; Owner: lbaw1653
--

CREATE TRIGGER concludedissue AFTER UPDATE OF "issueResolved" ON
"Issue" FOR EACH ROW EXECUTE PROCEDURE concludedissue();

--
-- Name: EquipmentSports_equipmentSportSportID_fkey; Type: FK
CONSTRAINT; Schema: public; Owner: lbaw1653
--

ALTER TABLE ONLY "EquipmentSports"
    ADD CONSTRAINT "EquipmentSports_equipmentSportSportID_fkey" FOREIGN
KEY ("equipmentSportsSportID") REFERENCES "Sport"("sportID");

--
-- Name: EquipmentSports_equipmentSportsEquipmentID_fkey; Type: FK
CONSTRAINT; Schema: public; Owner: lbaw1653
--
```



```
ALTER TABLE ONLY "EquipmentSports"
    ADD CONSTRAINT "EquipmentSports_equipmentsportsEquipmentID_fkey"
FOREIGN KEY ("equipmentSportsEquipmentID") REFERENCES
"Equipment"("equipmentID");

--
-- Name: Equipment_equipmentsportsEquipmentID_fkey; Type: FK CONSTRAINT;
Schema: public; Owner: lbaw1653
--

ALTER TABLE ONLY "Equipment"
    ADD CONSTRAINT "Equipment_equipmentsportsEquipmentID_fkey" FOREIGN KEY
("equipmentSportsEquipmentID") REFERENCES "Equipment"("equipmentID");

--
-- Name: Equipment_equipmentsportsEquipmentID_fkey; Type: FK CONSTRAINT;
Schema: public; Owner: lbaw1653
--

ALTER TABLE ONLY "Equipment"
    ADD CONSTRAINT "Equipment_equipmentsportsEquipmentID_fkey" FOREIGN KEY
("equipmentSportsEquipmentID") REFERENCES "Equipment"("equipmentID");

--
-- Name: Issue_issueRentalID_fkey; Type: FK CONSTRAINT; Schema: public;
Owner: lbaw1653
--

ALTER TABLE ONLY "Issue"
    ADD CONSTRAINT "Issue_issueRentalID_fkey" FOREIGN KEY
("issueRentalID") REFERENCES "Rental"("rentalID");

--
-- Name: Issue_issueRentalID_fkey; Type: FK CONSTRAINT; Schema: public;
Owner: lbaw1653
--

ALTER TABLE ONLY "Issue"
    ADD CONSTRAINT "Issue_issueRentalID_fkey" FOREIGN KEY
("issueRentalID") REFERENCES "Rental"("rentalID");

--
-- Name: Manager_managerComplexID_fkey; Type: FK CONSTRAINT; Schema:
public; Owner: lbaw1653
--

ALTER TABLE ONLY "Manager"
    ADD CONSTRAINT "Manager_managerComplexID_fkey" FOREIGN KEY
("managerComplexID") REFERENCES "SportsComplex"("complexID");

--
-- Name: Manager_managerComplexID_fkey; Type: FK CONSTRAINT; Schema:
public; Owner: lbaw1653
--

ALTER TABLE ONLY "Manager"
    ADD CONSTRAINT "Manager_managerComplexID_fkey" FOREIGN KEY
("managerComplexID") REFERENCES "SportsComplex"("complexID");

--
-- Name: Manager_managerID_fkey; Type: FK CONSTRAINT; Schema: public;
Owner: lbaw1653
--

ALTER TABLE ONLY "Manager"
    ADD CONSTRAINT "Manager_managerID_fkey" FOREIGN KEY ("managerID")
REFERENCES "User"("userID");

--
-- Name: Manager_managerID_fkey; Type: FK CONSTRAINT; Schema: public;
Owner: lbaw1653
--

ALTER TABLE ONLY "Manager"
    ADD CONSTRAINT "Manager_managerID_fkey" FOREIGN KEY ("managerID")
REFERENCES "User"("userID");

--
-- Name: RentalEquipment_rentalEquipmentEquipmentID_fkey; Type: FK
CONSTRAINT; Schema: public; Owner: lbaw1653
--
```

```
ALTER TABLE ONLY "RentalEquipment"
    ADD CONSTRAINT "RentalEquipment_rentalEquipmentEquipmentID_fkey"
FOREIGN KEY ("rentalEquipmentEquipmentID") REFERENCES
"Equipment"("equipmentID");

--
-- Name: RentalEquipment_rentalEquipmentRentalID_fkey; Type: FK
CONSTRAINT; Schema: public; Owner: lbaw1653
--

ALTER TABLE ONLY "RentalEquipment"
    ADD CONSTRAINT "RentalEquipment_rentalEquipmentRentalID_fkey"
FOREIGN KEY ("rentalEquipmentRentalID") REFERENCES
"Rental"("rentalID");

--
-- Name: Rental_rentalSpaceID_fkey; Type: FK CONSTRAINT; Schema:
public; Owner: lbaw1653
--

ALTER TABLE ONLY "Rental"
    ADD CONSTRAINT "Rental_rentalSpaceID_fkey" FOREIGN KEY
("rentalSpaceID") REFERENCES "Space"("spaceID");

--
-- Name: Rental_rentalUserID_fkey; Type: FK CONSTRAINT; Schema: public;
Owner: lbaw1653
--

ALTER TABLE ONLY "Rental"
    ADD CONSTRAINT "Rental_rentalUserID_fkey" FOREIGN KEY
("rentalUserID") REFERENCES "User"("userID");

--
-- Name: SpaceSports_spaceID_fkey; Type: FK CONSTRAINT; Schema: public;
Owner: lbaw1653
--

ALTER TABLE ONLY "SpaceSports"
    ADD CONSTRAINT "SpaceSports_spaceID_fkey" FOREIGN KEY
("spaceSportsSpaceID") REFERENCES "Space"("spaceID");

--
-- Name: SpaceSports_sportID_fkey; Type: FK CONSTRAINT; Schema: public;
Owner: lbaw1653
```

```
--  
  
ALTER TABLE ONLY "SpaceSports"  
    ADD CONSTRAINT "SpaceSports_sportID_fkey" FOREIGN KEY  
    ("spaceSportsSportID") REFERENCES "Sport"("sportID");  
  
--  
-- Name: Space_spaceComplexID_fkey; Type: FK CONSTRAINT; Schema:  
public; Owner: lbaw1653  
--  
  
ALTER TABLE ONLY "Space"  
    ADD CONSTRAINT "Space_spaceComplexID_fkey" FOREIGN KEY  
    ("spaceComplexID") REFERENCES "SportsComplex"("complexID");  
  
--  
-- Name: SportsComplex_complexMunicipalityID_fkey; Type: FK CONSTRAINT;  
Schema: public; Owner: lbaw1653  
--  
  
ALTER TABLE ONLY "SportsComplex"  
    ADD CONSTRAINT "SportsComplex_complexMunicipalityID_fkey" FOREIGN  
KEY ("complexMunicipalityID") REFERENCES  
"Municipality"("municipalityID");  
  
--  
-- Name: User_userMunicipalityID_fkey; Type: FK CONSTRAINT; Schema:  
public; Owner: lbaw1653  
--  
  
ALTER TABLE ONLY "User"  
    ADD CONSTRAINT "User_userMunicipalityID_fkey" FOREIGN KEY  
    ("userMunicipalityID") REFERENCES "Municipality"("municipalityID");  
  
--  
-- Name: public; Type: ACL; Schema: -; Owner: lbaw1653  
--  
  
REVOKE ALL ON SCHEMA public FROM PUBLIC;  
REVOKE ALL ON SCHEMA public FROM lbaw1653;  
GRANT ALL ON SCHEMA public TO lbaw1653;  
  
--  
-- PostgreSQL database dump complete  
--
```

## Revision History

Date	Description
2017/04/11	- Deleted equipment occupied
2017/05/24	- Full-text search reviewed
2017/06/1	- Full-text search reviewed with the indexes used in the final version - New SELECT's added in the frequent queries - New UPDATE's added in the frequent updates - SQL updated with the new version - SELECT's and UPDATE's already in the artifact were updated to their final version

From:

<http://lbaw.fe.up.pt/201617/> - **L B A W :: WORK**

Permanent link:

<http://lbaw.fe.up.pt/201617/doku.php/lbaw1653/a6>

Last update: **2017/06/01 17:30**

