1. PROBLEM DESCRIPTION

The Terrestrial Confederation of Physicians (identified by the acronym CONGATE) is an international ministry to which most of the health professionals on this planet belong. This institution wants to launch the *Congate-Cares* website, where citizens can find information about private health insurance and the companies that offer it. For its development, they have collected the information requirements that must be observed through the design of the DB that will support that website.

For each insurance company, it is required to register its commercial name, CIF (tax identifier), headquarters address, city, ZIP code, telephone, email, URL of its website and the information of the hospitals associated with that company.

The name of a hospital is unique and identifies the hospital throughout the world. Any hospital can have a concert with an insurance company, with several, or with none of them. In addition to the name of the hospital, we need to observe the hospital tax identifier code (CIF) and its address. Regarding addresses, we should record both the main entrance and its emergency service entrance. Nonetheless, both entries may coincide, and it is also possible that the hospital does not have an emergency service. Apart from that, there are some other characteristics related to hospitals, as the name of the city/town, ZIP code and telephone number (13 digits, different for each hospital). Collaboration contracts (hospital-insurance company) have start and end dates and cannot overlap in time. If a new contract is signed before a previous contract ends, the previous contract is immediately resolved (the end date of the former contract will be automatically assigned to the day before the new contract becomes effective, that is, the new contract's starting date minus one), so that the contracts do not overlap in time. In addition, any two contracts with the same parties (same hospital and company) cannot have the same starting date.

Each hospital has a range of specialties (orthopedics, dermatology, dentistry, etc.; there are currently 49 different ones). For each specialty, its name (identifier), a brief description, and the hospitals that are offering such medical services are to be recorded. At each hospital, each specialty can be supported by one or more physicians (or none, in case that hospital doesn't offer that specialty). For each doctor, it is required to keep the collegiate number (identifier), full name, citizen ID (DNI) / passport number (up to 15 characters), locator number and his/her specialties (at least one). Any doctor can only practice the specialty(s) in which he/she is a specialist, although there is no limit to the number of specialties held. If a hospital goes bankrupt, its doctors would be temporarily unemployed (unassigned).

Each insurance company may offer different types of insurance (products). Each of them will have its own set of specialty coverages. For example, VITASA's "Basic" insurance only covers the "family doctor" specialty, with a 0-day waiting period. The

"waiting period" is the stipulated time that must elapse from the contracting of the product until the client is effectively covered and can have the correspondent medical services. The coverages linked to a product cannot be removed or modified while the product is contracted by any customer. A new version of the product can be released, with different coverages, and this new version can become the active one, making the previous one obsolete so no longer it can be contracted.

For this web service, only basic personal data is collected from customers (DNI or passport, which is univocal; name, surname/s, gender, and personal email). In addition, it must allow collecting what product each client has contracted (or products, because they can have more than one), with start date, duration in days, number of people covered by the coverage, and medical appointments (past and future), so the platform will allow managing appointments at some time in the future. Appointments are defined as a visit by a patient (client) to a doctor in a hospital to handle a problem related to a specialty (held by that doctor), on a certain date and time.

The current database is extremely poor, with only four disjointed tables (and hardly any restrictions): a table registering doctors and their specialties; another one regarding accords between insurance companies and hospitals; another one including descriptions of products and their coverages; and the last one keeping the insurance policies (contracts between clients and insurance companies).

```
SQL> desc fsdb.doctors

Name                      Null?    Type
------------------------- -------- -------------
collegiateNum                      VARCHAR(12)
name                               VARCHAR(40)
surname1                           VARCHAR(25)
surname2                           VARCHAR(25)
passport                           VARCHAR(15)
phoneNum                           NUMBER(13)
specialty                          VARCHAR(50)
desc_specialty                     VARCHAR(150)
hospital                           VARCHAR(50)
address_hospital                   VARCHAR(50)
address_emergency                  VARCHAR(50)
ZIP_hospital                       NUMBER(5)
town_hospital                      VARCHAR(35)
country_hospital                   VARCHAR(50)
phone_hospital                     NUMBER(9)
```

```
SQL> desc fsdb.contracts

Name                          Null?    Type
----------------------------- -------- -------------
hospital                               VARCHAR(50)
address_hospital                       VARCHAR(50)
address_emergency                      VARCHAR(50)
ZIP_hospital                           NUMBER(5)
town_hospital                          VARCHAR(35)
country_hospital                       VARCHAR(50)
phone_hospital                         NUMBER(13)
insurer                                VARCHAR(40)
taxID_insurer                          VARCHAR(10)
address_insurer                        VARCHAR(50)
ZIP_insurer                            NUMBER(5)
town_insurer                           VARCHAR(35)
phone_insurer                          NUMBER(13)
email_insurer                          VARCHAR(30)
web_insurer                            VARCHAR(30)
start_date                             VARCHAR(10)
end_date                               VARCHAR(10)
```

```
SQL> desc fsdb.coverages

Name                          Null?    Type
----------------------------- -------- -------------
product                                VARCHAR(50)
version                                number(4,2)
launch                                 VARCHAR(10)
retired                                VARCHAR(10)
coverage                               VARCHAR(50)
description                            VARCHAR(150)
waiting_period                         VARCHAR(12)
insurer                                VARCHAR(40)
taxID_insurer                          VARCHAR(10)
address_insurer                        VARCHAR(50)
ZIP_insurer                            NUMBER(5)
town_insurer                           VARCHAR(35)
phone_insurer                          NUMBER(13)
email_insurer                          VARCHAR(60)
web_insurer                            VARCHAR(30)
```

```
SQL> desc fsdb.clients

Name                          Null?    Type
----------------------------- -------- -------------
passport                               VARCHAR(15)
name                                   VARCHAR(40)
surname1                               VARCHAR(25)
surname2                               VARCHAR(25)
gender                                 VARCHAR(6)
email                                  VARCHAR(60)
CIF_insurer                            VARCHAR(10)
product                                VARCHAR(50)
version                                NUMBER(4,2)
contracted                             VARCHAR(10)
duration                               NUMBER(4)
```

2. – SUPPORTING MATERIALS

Apart from classes and tutoring sessions, students can count on the following resources:

- Documents: assignment statement (this doc); class slides; template for writing the assignment report
- Audiovisual resources: video classes to acquire specific knowledge about the use of the tools that will be used in the laboratories (console management and pl/sql syntax) in the 'inverted class' style.
- Sw Resources: user account on RDBMS Oracle (accessible from all computer rooms in the University, and from Aula Virtual), with enough privileges for all required operations and reading privileges on the obsolete DB's tables.

3. – TO DO

a) Do an adequate Relational Design meeting the requirements (as possible). Represent it by means of a relational graph (use notation provided in class) and accompany with relevant comments about the non-observed explicit semantics (each assumption not reflected will have an identifier or a number to properly reference it in other sections), and the implicit semantic assumptions that have been included in the design.

b) Implement that design in PL/SQL (on DBMS Oracle®). The simple restrictions deemed appropriate will be included, indicating the explicit semantics they incorporate. Likewise, the new implicit/explicit-excluded semantic assumptions that appear in this phase will be documented.

c) Do a massive data upload (import) from the old DB to the newly created one. Notice there could appear errors during this process, due to the lack of some attributes and/or constraints (poor semantic coverage) in the former database. Coming from insufficiently defined tables, data may present inconsistencies, lack of integrity, and data errors in general. You must detect, analyze, typify and describe them in the report. If possible, propose and implement some solution.

d) Document all the work carried out by means of the pertinent *Labwork Report*, for which composition a template is provided. Apart from including the relational graph and semantic comments, make sure that all design decisions are conveniently justified and thus reflected in the report.

e) Submit through Aula Global a compressed .zip file containing three files: two scripts (creation.sql, upload.sql) and the *labwork report* saved as .pdf (portable document format), and named ***nia1_nia2_nia3_LW1.pdf***