DSIC
DEPARTAMENTO DE SISTEMAS INFORMÁTICOS Y COMPUTACIÓN

Escuela Técnica Superior de Ingeniería Informática

etsinf

Degree in Informatics Engineering

Databases and Information Systems

Conceptual Design

Exercises

UNIVERSITAT POLITÈCNICA DE VALÈNCIA

V. 21rev2

Table of Content

1. RESTAURANT

A restaurant wants to make its menus available through mobile devices. The information deals with menus, dishes, cooks, ingredients and wines. The restaurant offers several menus for which we have a code, a price and the dishes it is composed of (at least one dish in a menu). Each dish may be designed by a cook (but not more than one), with DNI (national identity number), name, country and age. Each dish has a code, its name and (optionally) its approximate number of calories. For some dishes we also want to store information about its ingredients (including their quantity) and the recommended wine for the dish. We will store the code of the wine, the type (red, white, or rosé), the name, and the vintage. Finally, for each ingredient we will store its code, name, price, and the stock. Having this information for each ingredient is mandatory.

2. CYCLING

A news website wants to create a new section to cover Le Tour de France, with complete results and statistics about the race. The readers will be able to get all the information about teams, cyclists (riders), climbs (mountain passes), jerseys (maillots) and stages.

For each team we want to store the name, the director (optional), and the cyclists in the team. All cyclists must belong to one and only one team.

For each cyclist we need their number (assigned to the cyclist for the whole race), name, age (optional), name of their team, stages they won, climbs (mountain passes) they have gone through in first place and the jerseys they have worn in each stage.

For each climb we could store the name, maximum height, category (“1st", "special",…), slope, stage where it is located (one and only one) and the cyclist who has passed it in first place.

For each jersey, we could save the jersey code, type, color, prize for wearing it and cyclists who have worn it.

For each stage we will consider the stage number, length of the stage in km, departure city, arrival city and the cyclist who has won the stage.

Two different cyclists cannot wear the same jersey during the same stage.

Since the database will be updated during the race, many stages and climbs will have no winner initially. Information about who is wearing the jerseys for each stage is also unknown until each stage is finished.

3. INSURANCE COMPANY

An insurance company wants to manage the “claim adjustments act” for insured vehicles that have had an accident. Each claim adjustment is identified by a reference number, and has a date, the vehicle, the assigned adjuster (code and name), and the data about the mechanic (name and address) where the adjustment has been made. Each vehicle is coded by a sequential identifier. For regular vehicles we also consider its license plate, for motorcycles we use the municipality license number and for any other vehicle (e.g., bicycles), an internal code. Plates, municipality numbers and internal codes must be (globally) unique. For each vehicle we are also interested in the make, the model, the owner and, if the vehicle is insured, the

insurance policy number, the company, the type of insurance and the expiration date. Every insurance policy is always linked to a vehicle. We will only consider information about vehicles which have had a claim adjustment.

We will also include information about the breakdown of the vehicle pieces, with their description (e.g., "XX1": "Front bumper", "XX2": "Left headlight", "XX3": "boot"). The result of a claim adjustment consists of an estimation of the parts of the vehicle which have been affected. For each affected part, the claim adjustment act issues a diagnosis which indicates whether it has to be repaired or changed, and the estimated workload (in hours).

Also, it details the materials that would be needed for the repairing of the affected part. We also have information about all these materials, with their code, description and price.

4. ART GALLERY

One art gallery wants to design a database to store all the information on the exhibitions and the paintings that the gallery sells. The most relevant information to be stored is described below.

There are several artists working for the gallery and we would like to store their ID, name, address, and phone number. These artists provide paintings for the gallery and the gallery organizes an exhibition to sell the paintings. Each painting is identified by a code, and it has a title and a price established by the painter. The exhibitions are identified by a code and they have an initial and a final date. The gallery also wants to store information about the customers to advise them of the next exhibitions: the customer ID, name, address, and phone number. An artist can also be a customer, so they will be advised of the exhibitions.

A painting can be exhibited in several exhibitions until it gets sold to a customer. The gallery wants to keep the information on which paintings have been included in each exhibition, the date of sale (note that this date can be different from the exhibition date), and the final price.

5. BOOK LIBRARY

We are interested in designing a database for managing a small library. After analyzing the system, the most frequent operations have been identified:

- Obtain the book information: book code, title, author (or authors), theme, and reader currently in possession of the book (if the book has been borrowed).
- Obtain the information about the readers: reader ID, name, address, phone number, and the books that the reader currently has and their loan dates.
- Add, remove, and update the readers' information.
- Manage the loans: Lend a book to a reader and record the return date.

Some integrity constraints have been identified:

- The book code uniquely identifies the book.
- The reader code uniquely identifies the reader.
- The themes used to classify the books are physics, electricity, mechanics, and optics.
- The total number of books borrowed by a reader is a derived attribute and it will be automatically updated by the system.

6. THE "PERIÓDICA" MAGAZINE

"Periódica" is a bi-weekly economic and political magazine. The magazine staff needs a database system to store the information of the "Periódica" issues, the articles included in each issue, and other related information described in the next paragraphs.

Each issue has an ID, a publishing date, and a circulation (expressed as number of copies). Each issue is divided into several sections (such as economy, currency, national politics, international politics, or investments) where articles are published. Each section is identified by a code and contains a description.

Each article is identified by a code, and it also has a title, an abstract, and the text of the article. When a new article is added, we need to know the author (his/her code, name, and phone number), and the section of the magazine in which the article will be included. After that, we will have to decide in which issue of the magazine the article will be published (an article can only appear in one issue).

The articles are divided into two categories depending on the author: It can be written by a member of the editorial staff (the article is called 'chronic' and has an attribute indicating the urgency of its publication), or by a freelance (the article is called 'collaboration' and it has an attribute indicating its score and the justification of that score).

Collaborations always arrive by email, so we want to keep track of the received emails. Each email will have an ID (number), date of entry, a sender, and a subject. A single text document is attached to the messages. This attachment can be of two types: It may be the original document, with the text of the collaboration, or it can be a companion document to update the original document of an existing collaboration. In the latter case, we need to know the type of update required: add text to the document, replace some text in the document, or delete the text that appears in the companion document. Each collaboration has one (and only one) original document but we can receive several companion documents.

7. CAR RENTAL

A car rental company wants to create a database to organize its activities.

The company has several cars. Each car has a plate number, chassis number, brand name, model, date of manufacture, mileage, and number of occupants. There are four categories of vehicles in the company: tourism, luxury, SUV, and truck. Depending on the vehicle category the company applies different rates.

Customers can be persons or companies (firms, government agencies, etc...). All customers have DNI/NIF, address, and phone number. People have a name (name and surname), and driving license antiquity. On the other hand, companies have a name and a contact person name.

The car rental company offers a chauffeur service. For each driver (chauffeur) we have his/her DNI number, name, address, telephone number, driving license date, social security number and salary. Any rental contract must indicate if the rental includes a chauffeur or not. If it is a chauffeur service we'll assign a driver. A driver can only be assigned to one rental contract that has not expired.

The rental contract contains the person or company that has rented the vehicle. Each contract has a contract number, pickup date (mandatory) and return date (planned and actual). The contract may include chauffeur and it must specify whether the customer will paid by kms, days or weeks.

The method of calculating the amount to pay is as follows:

- Num_km * Km_price_car_category
- Num_days * Daily_price_car_category
- Num_weeks * weekly_price_ car_category

Chauffeur service: additional fee of 60€/day (10,000 pesetas).

In case of delay in returning the vehicle, there will be a fee of 60 €/day (10,000 pesetas).

A bill will include one or more contracts corresponding to the same customer. It will have a number, a date and it will include all the rental contracts and the their total amount (in euros).

The most frequently queries and updating requirements are:

- List all terminated contracts indicating the rented vehicle and the customer information.
- Create invoices.
- List all vehicles available at any given time.
- List the rates for each type of vehicle.

8. A SUPERMARKET

A large grocery store has decided to create a home-delivery service. The supermarket has prepared a warehouse next to the store where purchases are deposited until the dealer picks up and distribute them. In this warehouse several cold storages have been installed. Each cold storage is uniquely coded and we want to know its scheduled maintenance date and the name and telephone number of the maintenance technician. The store also has installed some numbered shelves. We are interested in knowing know what the height of each shelf is. When a customer completes his/her purchase, if he/she is a new customer he/she must give his/her personal data (ID, name, phone number, and address) to the cashier. The cashier is responsible for distributing the purchase in bags labeled with a serial number and entering this information into the system along with the ticket number (which identifies a purchase) and the date.

The bags can be labeled as "keep in cold place" or not. The bags are stored in the warehouse in shelves or in cold storage. The manager of the warehouse saves where each bag is located. The delivery boy goes to the warehouse and pick up the ticket of a purchase. The ticket includes the position of all the bags of that purchase and who was the cashier that prepared this purchase.

It is very important to know the time that each purchase has been delivered and the delivery boy who delivered that purchase. This information is introduced by the delivery boy and will be used to calculate a gratification for the workers at the end of the month.

The system must store the worker ID, name, age and phone number of each delivery boy. The worker ID, name, and category of each cashier will be also stored.

9. THE UPV SAILING CLUB

We want to design a database to manage the UPV Sailing Club as a contact network. This database will be used to contact crew members of the UPV (staff, student and alumni) who want to participate in races, with boat owners looking for a crew for their boats. We want to know the school in which each student is enrolled.

For each boat we must store its characteristics, such as model, the name of the boat (that serves as ID), length, beam, yacht club, and owner. We need to store for each owner his/her full name, DNI, mailing address, and one or more phone numbers (one at least). Moreover, the UPV Sailing Club has information on all the races: the name of the race, its organizing yacht club, and the race category. The races are celebrated each year (different editions). For each race edition we know the start and end dates, and the number of available positions on each boat. This information is transmitted to the members of the UPV Sailing Club to assign crew to each of the ships.

We need to create a final report containing the members of the club who have participated in each race edition (in a specific year) and which ship has been used. For each member of the club we need to store his/her sailing license number, his/her name and address, phone number, if he/she is a student (enrolled in a school) or a member of the staff, and his/her sailing qualifications. We will also store the basic contact information of the yacht clubs that organize the races: name, initials of its name (for identifying purpose), address, phone number, and a brief description.

Other systems requirements are:

- List of boats owners (to send them mail).
- List the members of the UPV Sailing Club to send mail.
- List of boats that need crew for a specific race edition.
- How many participant has been in a race, the crew that has participate in a race, and the boats used.
- Boats offering places for a specific edition of a race.

10. PRIMARY SCHOOL

We want to design a primary school database.

Each course is identified by a number and a level. In any course there are several groups coded with a letter (A, B, C,…). Each group has always an assigned classroom, and several teachers (at least one). Students are assigned to one group as soon as they are enrolled (each student is assigned to a group and only one). One of the teachers lecturing in a group will be the group tutor. We will store the teacher ID, his/her name, and his specialty (if he has one). We need to store the following student information: name, registration number, DNI, name, address and a contact phone number (father, mother, or legal tutor). Some of the students have lunch in the school, and some students use the school bus. Of those students having lunch at school we have to store his/her lunch turn and his food restrictions. Of those students using the school bus we need to store a bus number and a bus top. We would also store the extracurricular activities. One student

can be enrolled in more than one activity. Activities have a code, a description and the days of the week in which this activity is scheduled.

Each course has a parent representative (a parent of one of the students enrolled in that course). Each parent can only be representative of a course. The school also has some material like sports equipment, musical instruments, art supplies,… All these elements are coded, and we will have to store their description and the number of units available. At the beginning of the course, each teacher receives the supporting material that he need for each of the groups in which he teaches. We need to store this information including the number of units of each element assigned to each group.

11. AQUARIUM PARK

We are going to design a database for a large aquarium park. There are several areas in the park with a unique name, a situation, and a size. Some of the areas are tanks with aquatic animals, and others have different functions (reception, restaurant, maintenance, etc.).

There is a species assigned to each area. We know the code (ID of the species) its scientific name (unique), the place of origin, the class of animal (mammal, bird, or fish) the number of animals, and the tank in which all the animals of that species live.

Some of the species are considered 'large'. We store individualized information for these animals: ID (a number unique in that species), a name (not repeated), if it was born in the park or outside, its incorporation date, and its weight.

We store information related to park workers: employee ID, name (required), DNI (which is unique) and salary. The staff is organized according to a hierarchical structure, so that each employee is subordinate to one (and only one) employee (his boss). We have to store the information of this hierarchy, and the date in which each boss was assigned to some worker. A person cannot be subordinate and boss of the same person (except one: the president, who is head of himself). Some workers are considered “animal caretaker” and they are in charge of one or several species.

We have to store the pricing table. This table includes the price for adults and children for holidays and business days.

12. A BUILDING COMPANY

A large construction company dedicated to the refurbishment of properties wants to create a database. We will store the employee number, which is unique, his/her DNI (which is also unique) the name, address, telephone number and his/her role in the company (architect, architectural technician or worker).

When a client requests a reform, all his/her data is stored: DNI (his/her identification), name, full address and phone number, and the data related to the refurbishment: a unique code, the refurbishment and the full address of the property to be refurbished. A refurbishment is assigned to an architect, who will be in charge of carrying out the refurbishment project. We also know the architect collegiate number (which is unique) and his/her specialty.

When the architect prepares a refurbishment project, he/she marks the refurbishment as ready, a copy of the project is registered, and an expected start date of the refurbishment is registered.

When a refurbishment starts, it is recorded as initiated; we store the start date and the architectural technician who will be responsible for the supervision. We have to store the architectural technician's mobile number in the database. Every day we have to generate a report for any active refurbishment. The workers who will be working that day must appear in this report, and for each of the workers, the task assigned. The tasks to be performed by the workers are classified and have a unique code and description. We have to store for each worker the tasks for which he/she is qualified. A worker may be assigned only a task for which he/she is qualified. Moreover, the daily report also includes a list of the materials to be used that day, and the amount of each one of them. There is a record of the materials (with a unique code), a description, and their stocks.

At the end of the day, each worker must tell the architectural technician the incidences that has occurred in the tasks that he/she was assigned. This information is recorded in the daily report.

When a refurbishment is finished, the end date is stored in the database.

13. A SCRAP CAR COMPANY

A scrap car company wants to create a new database. We have the following description of the company:

The cars normally arrive at the company in one of its cranes. Then the company decides if the car can be repaired or is going to be scrapped. If the car is going to be repaired, a mechanic is assigned.

Each vehicle is identified by its license plate and has several required attributes: year of manufacture, model, price paid, and the date of entry into the company. If a vehicle arrives at the company several times (for example the company repairs the car, then the company sells it, and later it re-enters for scrap), we only hold information about its last arrival.

In the company, there are workers who repair cars (mechanics), other employees who drive the cranes (drivers), and there are other employees who do other tasks. We need to store the DNI of all the employees, their social security number (unique), name, address, telephone number, and salary. There is a bonus for the drivers depending on how many cars they have brought in, so it is necessary to know that information. It is also important to know what crane has brought in each car. We need to know the crane identifier, the year it was purchased, the last mechanic that checked the crane and the date of the check. Each crane is checked when it is purchased, and on subsequent occasions.

We need to know the date on which the company has finished the scrapping of any car and the kilos of recyclable metal that have been obtained. Moreover, the company obtains parts to be reused. The company assigns numbers sequentially (starting from number one for each car) to those parts. We need to store a brief description and a price for each obtained part.

For every repaired car we need to know its final price, the date on which the car was repaired and the parking lot number in which it will stay until sold. A number identifies the parking lots for sale.

14. TELEPHONE CALL CONTROL

A large company has decided to create a computer system to control its phone calls. The company has signed contracts with various telephone operators. Each operator offers only one type of service: landline calls, mobile calls, and cable services. We need to know the NIF of the telephone operator (which identifies the operator), its brand name (which is unique), and its full address. Each landline operator has a prefix to dial and a set of toll free telephone numbers. Each cable provider offers a flat-rate internet service with a fixed price. Finally, we need to store a table with the discount offered on mobile calls depending on the day of the week.

For each call, we need to store its date (mandatory), source phone number, destination phone number, duration, operator, type of call, and time slot. All calls are coded with a number that is unique to each operator. The different types of calls are encoded. The call also has a name (or type of call: internal, local, long distance, international, ...) and a description. The different time slots (e.g., from 18h to 8h, 8h to 14h, ...) are common to all operators, and are also stored with their code and description.

We have a table with the rates of the operators including the price per minute for each type of call (local long distance, international,…).

In order to replace internal calls by email, we are going to monitor all the internal calls. We need to store the DNI and the name of the employee who has made and received internal calls.

15. THE "SOFTWWW" COMPANY

The "SoftWWW" company develops software for large companies. When a company contracts its services, it formalizes a contract. The contract contains a code, the date of the contract, the NIF and registered office of the company, and the project requirements. A team of company experts meets to study the contract and plan the project. Each project has a code, a start and end date, an analyst, and a set of tasks to be performed. Each task is identified by a sequential number within each project, and we store its description, the estimated time of completion, and the number of assigned programmers. Some tasks require the use of special software that is installed in some computers. In these cases, a computer is reserved for a programmer who will be working on one of these tasks for a period of time. The computers are properly coded and have a name and a location. We have the following employee data: DNI, name, address, telephone number, and job (analyst, programmer, administrative, etc..).

16. AIRLINE COMPANY

A small airline company wants to create a database on what is known as "generic flights", which are daily flights between two cities always at the same time and with a specific code. There may be more than one generic flight between the same cities but at different time.

On a particular day, for each flight, we need to know the assigned aircraft, the origin and destination airport, and the flight capacity. In addition, we must store the information about the reservation of seats: passenger list (a name is enough), number of available seats, and the assigned crew. Please note that there will always be at least one pilot in command and one co-pilot. We need the following data about the crew: employee ID, name, category, and a phone number (required).

The company has a fleet of aircrafts. The information for each aircraft includes the year of manufacture, the plate number, and the number of seats. We are also going to store information about the cities with airports, such as their name, country, the average temperature for each season, the city's population, the name of its airports (which is unique), its category, and the year of inauguration.

The most frequent queries and update requirements are:

- List of all flights between two cities.
- List of all flights between two airports.
- List of the crew assigned to a flight on a particular day.
- List of passengers on a flight on a particular day.

17. FINANCIAL CONTROL

The Ministry of Finance has decided to design a database to control the finances of the country. The information we want to store is detailed below.

We have several banks operating in the country. Each bank has a unique code, a name, and an address of its headquarters. Banks have branches (one at least) in different cities. Each branch has a branch number, which is unique for each bank, an address, and the name of the branch manager.

A citizen has a social security number, which is unique, a DNI, which is also unique, a name, an address, and a phone number.

A citizen can open one or more individual accounts, or accounts with other citizens. Each account is opened in a branch and is identified by an account number that is unique in that branch. We also want to know the current balance of the account, its opening date, and the applied interest rate.

We want to store information about the loans that each branch has granted to different people. Each loan is granted by a single branch and is identified by a loan number that is unique to that branch. Other information of interest is: the date, the amount awarded, the interest rate, and the citizen or citizens who have been granted the loan. Finally, we want to know the account number in which the loan bills are charged. Obviously, it must be the account of one of the citizens who have received the loan.

Banks can offer various types of investments (funds, time deposits, etc.) in which citizens can invest their money. Each type of investment has a unique code, a description, and a name. A citizen can make different types of investments, but each investment must be associated with one of his/her accounts and is identified by a unique code, and has a date and amount.

18. A SHELTER FOR STRAY DOGS

We want to create the database of a shelter for abandoned (stray) dogs. Each animal is registered with an identifying number. We want to store the date of arrival of each dog, the approximate date of birth (month and year), its breed, hair color, gender, and, sometimes, a short description. Each dog will also have a name.

The center has several cages where animals are assigned during their stay in the center. Each of these cages has an identification code, a capacity, and the date of the next fumigation.

When animals arrive at the center they are checked by a veterinarian and if there is no evidence that they are vaccinated, the dogs receive the vaccines corresponding to their age. We store the name of each vaccine (used to identify it), the virus of the vaccine, a description of the side effects, and the period of revaccination. Vaccines are bought in packages, each package has a lot number, which is unique for each vaccine, and an expiration date. When we apply a vaccine, we want to save the date, the dose that was used and the lot number of the applied vaccine.

There are people interested in adopting dogs. These benefactors are recorded with an internal code, their names, DNI, address and, at least one phone number.

Dogs can be in the center or adopted. A dog can be temporarily or permanently adopted. If the dog is temporarily adopted, we should store the date of delivery and the person who adopts the dog. When a dog is adopted permanently, we must also store the code of its identifying chip, and the name of the veterinarian who will have to castrate the dog. We only need to know the current status of the dog, no matter if it was previously adopted.

19. AN ELECTRICITY COMPANY

An electricity company has decided to develop a database to meet the large number of complaints received about the blackouts in recent days.

The company has contracts with its customers. A contract is identified by a number and it contains the date signed and the contracted capacity. A contract is to supply electricity to a single house. Each house only has one contract, and is identified by the street and number (unique in that city). A city is identified by a code, and it has the name of its province and the number of inhabitants of the city. A contract can be signed by a person or a company and contains the customer data: customer ID and address. If the customer is a person, the contract contains the DNI (which is unique) and the customer name. If the customer is a company, the contract contains the company NIF (which is unique) and its trade name.

When attending a claim, it is encoded and we take note of the date and details of the house making the claim. Each house can only make one claim. There are claims of two types: ordinary and extraordinary. The ordinary claims can obtain a compensation for damages. We need a list of all the possible damages, with their code, description and corresponding compensation. Each ordinary claim must contain the specific damages claimed.

Extraordinary claims do not cover the damages offered by the company but are accompanied by a list of damages. The damages in a claim are numbered sequentially, and all of them have their corresponding description and requested compensation.

The company has information on different power plants. Each power plant has a code, a name and a capacity. Furthermore, it is known which cities are supplied by each power plant (note that some large cities are supplied by more than one power plant).

Finally, for each power plant that supplies electricity to a city, we also have a list of time slots and days on which the city suffered a blackout.

20. A WINE COOPERATIVE

We have to create a database for a cooperative that sells bottled wines from different Spanish regions. Each wine is distinguished by a name, the year of production and type (red, rosé and white). We also have to know the region of origin (identified by its name) and the price of the wine. Each wine name always has the same origin. In each region, we obtain wine of different quality depending on the year. The quality is graded as poor, fair, good, very good or excellent.

The cooperative sells to partners. We know the partner name, code, address, city, state and zip code, and the partner number.

In addition to wine, which is the main product, the cooperative sells other food products. These products have a different name, a form of packaging, a weight, and a price. All products have a code that is unique.

Each partner can place orders that can include any combination of products, indicating a quantity for each product. An order has a date, an identification number, a total amount and a final amount (which is obtained by applying a discount rate). There is a discount table containing the percentage of discount depending on the total amount of euros of the order.

21. UPV RECTOR ELECTION

The “Universitat Politècnica de València” wants to create a database to manage the elections to UPV chancellor (rector). The census is composed of people belonging to the groups of teachers, students or administrative staff (PAS).

Each member of the census is identified by a DNI (ID) and also has a name, the group to which he/she belongs, and the center where she/he works. Each center is identified by a code and has a name and a size (in square meters).

There are several lists of candidates. Each list is identified by a unique code. We know the date of creation of the list and the members who compose it. All members of a list must be in the census and we want to know the position of the member in the candidate list. A member of the census cannot appear twice in a list or in more than one list.

The vote will take place in polling stations. Each polling station in a center is identified by a unique number (unique in that center). Each member of the census needs to know in which polling station and center he/she will vote.

Four members of the census are the chairs of each polling station. One of them is the president of the polling station. A person can only be member of one polling station.

On a voluntary basis, a candidate list may propose an auditor for each polling station. This auditor must be part of the census and can be an auditor in several polling stations. The same person cannot be an auditor for more than one candidate list.

22. ST. JAMES'S WAY (CAMINO DE SANTIAGO)

We need to create a database about the different routes of St. James's Way (Camino de Santiago). We want to store information about the different routes (the French Way, the Aragonese Way, the Northern Way, etc.).

For each route, we want to record its name, which is unique, its description and the number of total kilometers. Each route consists of a set of stages. These are identified by a sequential number within each route. For each stage we know the length in kilometers and some useful recommendations for the pilgrims.

Each stage runs through a set of locations. We must record this set of locations and the order in which they appear within the stage. We must be aware that a location can appear in several stages (of different routes) but a single route cannot pass twice through the same location.

Locations are identified by a code, and have a name. We also store their interesting tourist sites. For each site of interest, which is identified with a unique number within each location, we also store its name and a description.

For those locations (localities) with pilgrim's hostels, we want to know the number of inhabitants and hostels. The hostels are identified by a code, and have a name, an address, and a price.

Some locations have an office where pilgrims can stamp the pilgrim's pass(port), which certifies that the pilgrim has been in that location. For each office we want to store its name and its full address. We also want to have a record of pilgrims seeking certification. Each pilgrim has an identifier, a name, and address. We also record the locations she/he is certified to have visited, also indicating the date.

23. TRAFFIC

Nopacekistan (a country) is divided into traffic zones known as sections. The country's traffic police want to create an information system to have a better control of traffic fines and accidents. In order to do so, the police will store information about citizens, including their identifier, name and address.

The traffic police unit is composed of officers who always belong to one section. For each officer, we know their collar number, which identifies them, their name and address.

For each section, we will have an identifier, the number of inhabitants, the vehicles that pay their taxes inside that section and, finally, some information about fines and accidents.

Every fine and every accident always takes place in one section.

Fines are identified (within a section) by a number. Accidents are coded with a sequential number, re-initialised every day.

For each accident, we will have its date, time and the place where it happened, the officer that handled it and the vehicles (at least one) that were involved, also indicating the state of each vehicle after the accident. Each vehicle's driver will also be stored whenever possible.

Fines are issued to a specific vehicle, also specifying the place, the kind of penalty, the amount and the officer who issued the fine.

Finally, each vehicle is identified by its license number (plate), and has a vehicle type (car, motorcycle,…), model and colour.