# **CLASS DIAGRAM**

The class diagram describes the static architecture of a system using classes and the relationship existing among them. In our system the main class is the “User”, a generalization of the “Patient”, “Technical Administrator” and “Specialized Practitioner” children classes. It is characterized by different attributes and a single method, which are all inherited by these three specifications.

The children classes have also specific attributes and methods which are not valid for all the users. In particular, the “Patient” class has also an additional attribute, which is the *fiscal code*.

The other classes represent additional relevant objects of the system since they are the basis of the interaction among the different users. The first one is the “Visit” class, composed of the *visit ID*, *date*, *time* and *structure*. It establishes simple logical connections with the “Patient” and “Specialized Practitioner” classes through association links. The cardinality has been associated in the following manner: on the one hand, a patient can book “from 0 to many” visits and a visit can be booked by “exactly 1” patient; on the other hand, a specialized practitioner can make “from 0 to many” visits and a single visit can be made by “from 1 to many” specialized practitioners. The patient-visit link includes an association class (“BookingConfirmation”) to add some extra information to the two classes: it records the visit information, such as the *ID* of the visit, the *price* and the *booking code.* The specialized practitioner-visit link includes the “Visit Measurement” association class, in which all the measurements, collected during the visit, are registered. It is an efficient way to store all the data in the database.

Another class is the one called “Survey” which contains different attributes: *scores* as numerical evaluations, *date* of the submission, *structure* and *name* and *surname* of the *specialized practitioner* targets of the questionnaire and the *comments* that can be eventually added by the patient. The “Survey” class has a relationship either with the “Patient” or with the “Technical Administrator”. Each survey can be compiled by “1 to many” patients, and each patient can compile “from 0 to many” surveys. As far as the second relationship cardinality is concerned, each survey can be created by “exactly 1” technical administrator while a technical administrator can create “from 1 to many” surveys

The “Risk evaluation model” class contains a set of parameters and their thresholds. It is related to the “Specialized Practitioner” and the “Technical Administrator” classes. A specialized practitioner can visualize “exactly 1” risk evaluation model per patient, while a risk evaluation model can be visualized by “from 1 to many” SP (assumption: since the model is required by the SP, at least one SP will surely visualize it). This model is created by “exactly 1” technical administrator, as well as a technical administrator can create “from 1 to many” models.

Another class is the one of “My Measurements”, a generalization of other two classes: “Daily Measurements” and “Weekly Measurements”. In the subclasses there are some additional attributes which represent the specific parameters that have to be inserted daily or weekly into the system.

In the end there are other two classes, “ListOfPatient” and “ListOfUsers”, which are linked with a composition relationship respectively to the “Specialized Practitioner” and “Technical Administrator” classes.

A specialized practitioner can have “exactly 1” list of patients and a list of patients can be owned by “exactly 1” specialized practitioner. Instead, a technical administrator can have “exactly 1” list of users and a list of users can be owned by “from 1 to many” technical administrators.