CS 548—Fall 2017
Enterprise Software Architecture and Design
Assignment Two—XML Schemas

Design the XML Schema for a medical information system. You should provide the schema definition itself, structured as a collection of namespaces for the different form of documents, as well as instance documents that exhaustively demonstrate the different forms of documents that may be in such a system.

A clinic database consists of a collection of patient records, one for each patient. Each patient has:
1. a patient identifier (that may be assigned for example by a national government),
2. a patient name, and
3. a date of birth.

Furthermore each patient is related by a one-to-many relationship to a collection of treatment records.

Every treatment record includes a diagnosis of the condition for which the treatment is prescribed (e.g. throat cancer, HIV/AIDS, hepatitis, etc). Currently there are three specific forms of treatment records:
1. A drug treatment record includes a drug, a diagnosis, a prescribing physician, and a dosage.
2. A surgery treatment record includes a diagnosis, a surgeon and a date of surgery.
3. A radiology treatment record includes a diagnosis, a radiologist and a list of dates of radiology treatments.

The provider administering the treatment (as well as the diagnosis) can be a common field for all types of treatment records, specified by a provider identifier in the treatment record.

Every healthcare provider has a record that includes their provider identifier (NPI), their name, and an indication of their specialization (surgery, radiology, oncology, etc).

Define XML schemae for these entity types. Define:
1. A schema for identifier types, that defines patient identifiers and provider identifiers (as simple types, e.g., string).
2. A schema for treatment records, that defines a (complex) type that describes the different possible forms of treatments. There are several ways to do this, and you will be asked for solutions that explore each of these alternatives. Each treatment element will have at least a provider identifier and a description of the diagnosis.
3. A schema for patient records, that imports the schema for identifier types and the schema for treatment records. *The one-to-many relationship from patients to treatments is represented by embedding treatment records in patient records, so each patient element includes a list of treatments as child elements.*
4. A schema for provider records, that imports the schema for identifier types.
5. A schema for a clinic database, that imports the schemas for patient records and for provider records. A clinic database is simply a root element that contains patient and provider records.

There are three different forms of solutions, as described below. For each solution, provide example instance documents for a clinic database. If you have modeled the data as specified, the forms of the solutions will be different in each case.

For your first solution, use the *choice element* of XML schemae to represent the choice among the different forms of treatment records. Define a generic treatment record that specifies the diagnosis (and the administering provider), and use the choice element to represent the different forms of information that are recorded for the different types of treatment.

For your second solution, use *type substitutability* to represent the choice among the different forms of treatment records. Define a generic treatment record whose content specifies the diagnosis (and provider), and use derived complex types to represent the different forms of information that are recorded for the different types of treatment.

For your third solution, use *element substitution* to represent the choice among the different forms of treatment records. Define a generic treatment record whose content specifies the diagnosis (and provider), and use different elements in a common substitution group to represent the different forms of information that are recorded for the different types of treatment.

Finally, provide a UML class diagram that represents the relationships between the different entity types in this scenario. There are entity types for patients, providers and treatments. A patient may be on more than one treatment, including more than one drug treatment. Treatment types should be treated as reified relationship types (association classes), with a base class for general treatment and subclasses for the specific forms of treatment.

You should use the Oxygen XML Schema editor to edit the XML schemas and the XML instance documents[1]. **Do not generate the XML schemae from the UML class diagram.** It is possible using Eclipse to generate XML schemae, and an XML editor specific to an XML schema, from a UML class diagram. However the XML schema that is generated is too complicated for manual use, it is only useful in the context of model-driven architecture (MDA), where the schemae are only used by generated data entry and validation tools. But this is not part of the assignment.

Provide three different complete solutions to the problem, in three different folders for your solution. Provide example instance documents with each solution. Provide your UML class diagram in the root folder of your submission. One UML class diagram should suffice for all your solutions, since UML abstracts away from how you represent generalization and specialization.

Your solution should be uploaded via the Canvas classroom, as a zip file. This zip file should have the same name as your Canvas userid. It should unzip to a folder with this same name, which should contain the files and subfolders with your submission.

---

[1] Oxygen requires a license to use, but you can obtain a free 30-day trial license. Oxygen will be useful for this and the next assignment. You can also use Oxygen to generate sample data, but e.g., it will provide random strings for patient and provider names, so you are better off providing your own sample data.

It is important that you provide a document that documents your submission, included as PDF document in your submission root folder. Name this document README.pdf. This document should explain where to find the schema and instance documents for your submission, an overview of your schema and data model (including the UML class diagram), and any observations you have to make about design decisions you made during the assignment.