Project Brief

The aim of our ontology is to comprehensively describe a tool that helps students picking their future courses. The tool will take into account factors such as: topic overlap between courses, time schedules, previously taken courses, teacher/course-mate/course-schedule preference.

Classes: Person, Research Methodology, Period, Weekday, Course, Teacher, Student, Topic

Relations: *takes(Student, Course), teaches(Teacher, Course), hasTaken(Student, Course), uses(Course, ResearchMethodology), isPreliminary(Course, Course), covers(Course, Topic), isFriendWith(Student, Student).*
-The relation *isTaughtOn()* has two sub relations: *isTaughtOnWeekday(Weekday, Teacher)* and *isTaughtOnPeriod(Period, Teacher).*
-The relation *teachesDuring()* has two sub relations: *teachesDuringWeekday(Teacher, Weekday)* and *teachesDuringPeriod(Teacher, Period).*

Apart from the above, we will consider two more features belonging to our ontology:

- Class(Hobbies): students won't be able to attend a certain course based on their own time schedule. This might imply student having sports and/or extracurricular activities.

- Data Property(ECTS): each course is going to account for a certain number of credits. The total number of credits is going to be calculated by courses already taken and being taken during the period. A total and maximum amount of 15 credits per period should be reached.

In the following we elaborate on non-trivial restrictions which we are going to include in the knowledge base:

- A course covers at least one topic. It is taught by at least one teacher and uses exactly one research methodology. It is taught on exactly 2 weekdays.
  $Course \equiv \exists\ covers.Topic \sqcap \exists\ isTaughtBy.Teacher \sqcap = 1\ uses.ResearchMethodologies \sqcap = 2\ isTaughtOn.Weekday$
  $Course \equiv ScienceCourse \sqcup HumanitiesCourse$

- A teacher is a Person who teaches at least 1 course per period.
  $Teacher \equiv Person \sqcap \leq 1\ teaches.(\forall isTaughtOnPeriod.\{P1\})$
  $Teacher \equiv Person \sqcap \leq 1\ teaches.(\forall isTaughtOnPeriod.\{P2\})$
  ...

- A student is a Person who takes 2 or 3 courses per period
  $Student \equiv Person \sqcap \forall\ learnsDuring.Period \sqcap \geq 2\ takes.Course \sqcap \leq 3\ takes.Course$

Chance of a course to be preferred or a teacher not to be liked can't be represented in the ontology. Hence, using the different relations and data from each entity, is going to be up to the agent to infer how much the student likes a course or a lecturer.
For instance, a student that has taken courseX and courseY would be more likely to prefer courseZ rather than courseK because courseZ-X-Y have more topics in common.The latter reasoning is going to be computed in the agent system and not in the ontology itself.

Student data:

- Hobbies
- Topic preference
- Taken courses
- Is taking
- Dislike teacher
- Period

Queries:

Scenario 1:
Student A has completed period 1. He started playing football every thursday. From his previous education he knows that he prefers Natural Sciences courses. So far he has taken a Math and Machine Learning course in period 1.

Manchester Syntax:
Student A: *Covers ONLY NaturalScience AND NOT isTaughtOnWeekday ONLY Thursday AND isTaughtOnPeriod ONLY P2 AND (hasPreliminary ONLY Math OR hasPreliminary ONLY MachineLearning OR NOT hasPreliminary SOME Course)*

Scenario 2:
Student B is in his second year. So far, he had several courses in Natural Sciences but he wants to combine his knowledge with Topics in Humanities and Natural Science.

Manchester Syntax:
Student B: *Covers ONLY Humanities AND Covers ONLY NaturalScience AND isTaughtOnPeriod ONLY (P3 OR P4)*

Scenario 3:
Student C hates math, wants to take the same classes as his friend Mary (implies relation hasFriend(Mary)), doesn't like humanistic courses and wants to take Advanced Machine Learning next period which has Machine Learning 101 as preliminary.

Manchester Syntax:
Student C: *NOT Covers ONLY Math AND isTakenBy ONLY Mary AND NOT Covers ONLY Humanities AND isPreliminary ONLY AdvancedMachineLearning*