Camellia Guan (21026514)

COMP 3311: Task 1, Design Assumptions and Reasoning

Outline

Entities

- ➔ Faculty
  - Supervisor
  - Reader
  - Coordinator
- ➔ FYP
- ➔ Student
- ➔ ProjectGroup
- ➔ Grade

Relationships

- ➔ CreatedBy
- ➔ SupervisedBy
- ➔ GradedBy
- ➔ AssignedTo
- ➔ IsPartOf
- ➔ BelongsTo

Entities

Faculty

I represent faculty members in the entity called **Faculty**. Faculty has several required attributes that correspond with required faculty information. Faculty has three sub-entities that refer to the different types of faculty members (supervisor, reader, coordinator). These faculty members do not have any extra attributes, but they have different relationships with other entities, and this is represented through the connections to other entities. Coordinator is a type of Faculty, but no static information about Coordinator is specified in the project description, so there are no relationships between Coordinator and any other entity. An instance of Faculty is *partially complete and overlapping*, as I will assume that for example, a faculty member can simultaneously be a reader and supervisor (for different FYPs).

FYP

The final year project is represented in the entity called **FYP**. FYP has required attributes that were specified in the project description. The attributes include **{requirements}**, which is a multi-valued attribute that represents any other requirements that an FYP might have. Another additional attribute is **capacity ()**, which is a derived attribute that depends on the **type** of FYP, as a project has a maximum capacity of 4, and a thesis has a maximum capacity of 1. FYP is a weak entity.

Student

Students are represented through the entity called **Student**. Student, like Faculty and FYP, has required attributes that were specified in the project description. Even though the project description states that only 1 student can be assigned to a thesis and 4 can be assigned to a project, I do not represent this constraint in the E-R diagram because students should be assigned to project groups first, which then can deal with any other constraints.

ProjectGroup

Students must be assigned to a project group prior to indicating interest in an FYP, so I represent the project groups in the entity called **ProjectGroup**. ProjectGroup receives a code that is derived from the Faculty attribute, code. Therefore, I represent this derived attribute as **projCode ()**.

Grade

Because there are several graded requirements, I represent these requirements in an entity called **Grade**. None of the attributes are mandatory because there are initially no grades. Grade is a weak entity because of this.

Relationships

CreatedBy

It is not clearly specified whether or not a faculty member can create more than one FYP, but the project description states, “assigned to a faculty’s FYPs”, where FYP is plural. I then assumed that a faculty member can in fact create more than 1 FYP. Therefore, a supervisor can create 1 to many FYPs (we assume that they have to create at least 1 because they are a supervisor), and an FYP can be created by a faculty member. This is an identifying relationship because FYP does not have a primary key, and it holds a one-to-many relationship with Supervisor.

SupervisedBy

A supervisor can supervise 0 to 5 project groups, and a project group can have 1 to 2 supervisors.

GradedBy

A reader can grade 0 to many project groups, and a project group can be graded by at most 1 reader.

AssignedTo

A project group can be assigned to at least and at most 1 FYP, and an FYP can be assigned to 0 to many project groups.

IsPartOf

A project group can have from 1 to 4 students, while a student can be part of one, and only one, project group.

BelongsTo

A grade belongs to one and only one student, while a student can have two sets of grades (since both a supervisor and a reader need to grade a student). This is an identifying relationship because Grade does not have a primary key, and it holds a one-to-many relationship with Student.