TeacherApp

Use cases

- A teacher can log in in the TeacherApp.
- A teacher can create new tasks for a subject from the TeaherApp.
- A teacher can create new grades for a given student in a subject from the TeacherApp. The same student may have several different grades for the same subject.
- A teacher can create new meetings with parents of one of the students who attend their subjects.
- A teacher can change its own information (ie, change their password).
- A teacher can change the contents of the already created tasks in their subjects.
- A teacher can change the date of any of their already existing meetings.
- A teacher can change the grade of any existing grade in any of their subjects.
- A teacher can view their subjects, the students enrolled in them, and the parents of those students.
- A teacher can view the tasks that already exist in any of their subjects, the answers that the task has received, and the student who has sent each response.
- A teacher can view all the grades in any of their subjects, and the students to whom the grade corresponds.
- A teacher can check the meetings they have requested, and the parent with whom they are associated.

Functional Requirements

- The TeacherApp must be able to work offline as much as possible.
- To do this, it has its own domain model that will be located on the client.
- When a teacher logs in, the instances of all the existing elements on the server that are relevant to enable the use cases described above are brought through a query.
- If disconnected, the TeacherApp can still show the last updated information
- If disconnected, the TeacherApp can create the new objects necessary to enable the described use cases. When connected again, those instances will be uploaded.

Non-Functional Requirements

- All the functionality described above can be achieved by replicating the entire Global Domain Model (GDM). However, it would be a needlessly complex model for the requirements of the application.
    - Therefore, the Local Domain Model LDM must be as simple as possible.
    - Any classes, relationships, associations or navigations from the GDM that are not necessary for providing the requested functionalities can be removed.
    - At the same time, it must not lack the essential features required to support all the described use cases.