SWEN30006 Software Modelling and Design
Workshop 2: Domain Modelling and SSDs – Partial Solution

School of Computing and Information Systems
University of Melbourne
Semester 2, 2020

Disclaimer: The solution provided in this document is one of many possible solutions. Your solution may also be correct or suitable even though it looks different from the solution in this document. This is because the solution can be developed based on different reasoning and design decisions. In addition, the solution in this document may be *incomplete* for the purpose of learning. We encourage you to discuss with the tutor during the workshop hours to check the correctness of your solution.

Part 1 Static Domain Modelling

Q1.1 Interpreting a Domain Model

The biggest issues are the use of design elements (visibility modifiers[1], methods, navigation) and software concepts (types such as String and NOSQL DB Adapter) in the domain model. These are things that belong in a design model (covered later in the subject). Figure 1 shows where these issues can be found.

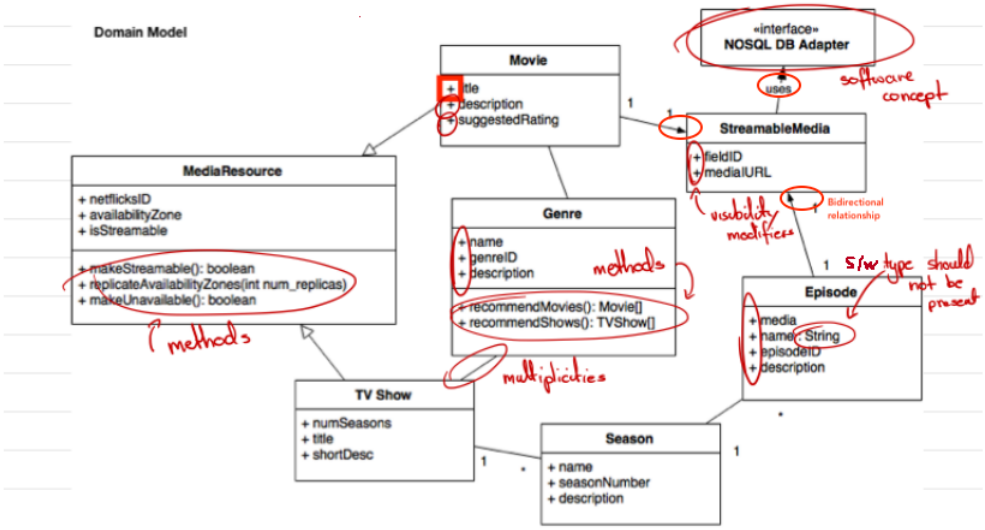

Figure 1: Issues with the domain model

[1]Some tools do not let you turn off visibility modifiers. If this is the case, it is best to have them all set to the same visibility (whether private or public) as in this example, and in the domain model below. Either way, they are not meaningful in a domain model.

Q1.2 Creating a Static Domain Model

You should end up with a single domain class diagram which looks reasonably similar to the one in Figure 2. It's okay if you have a differences in your diagram as long as you can explain your choices. Watch out for any software concepts – you should definitely not include these.

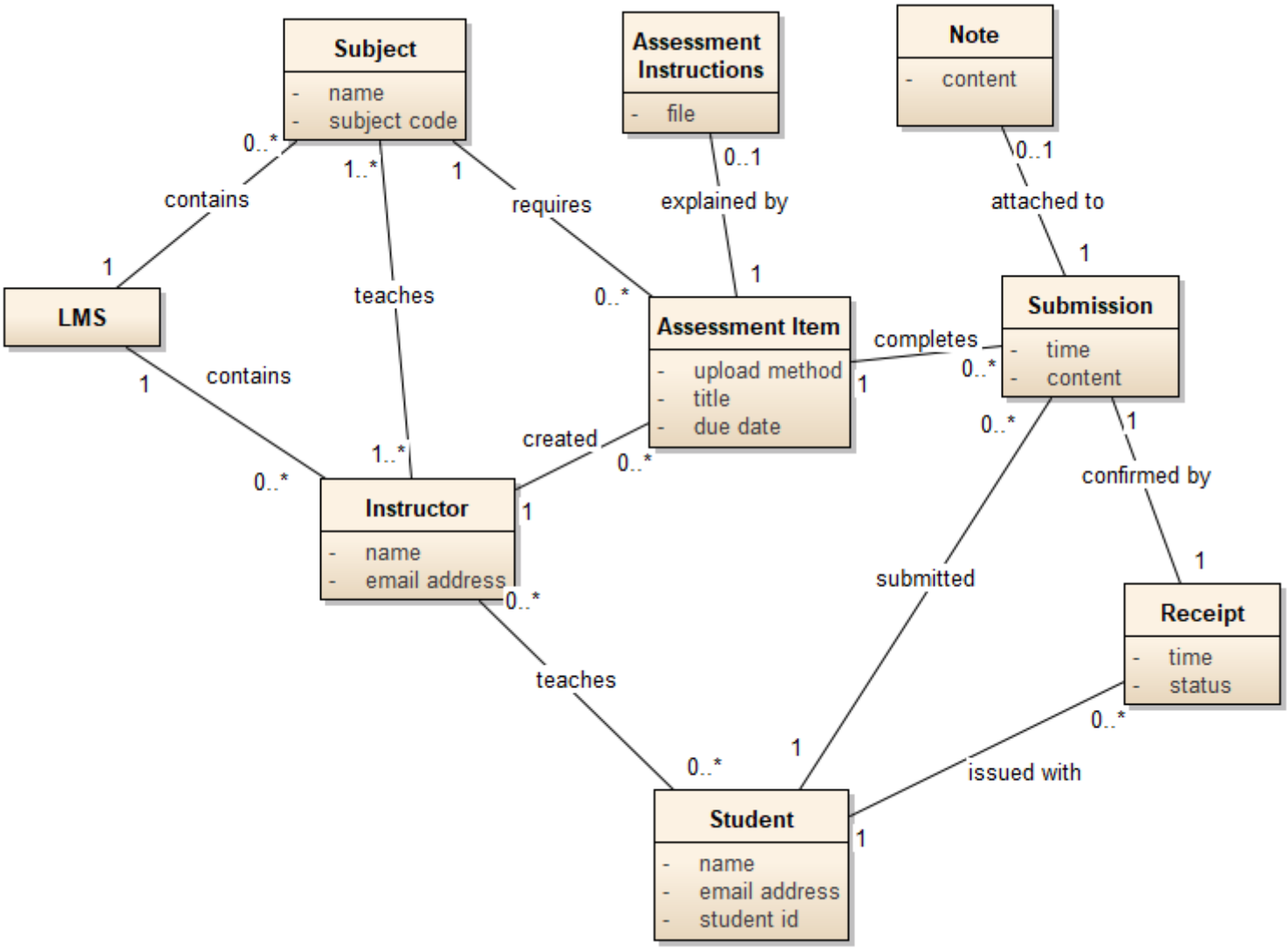

Figure 2: A complete domain model for Q1.2

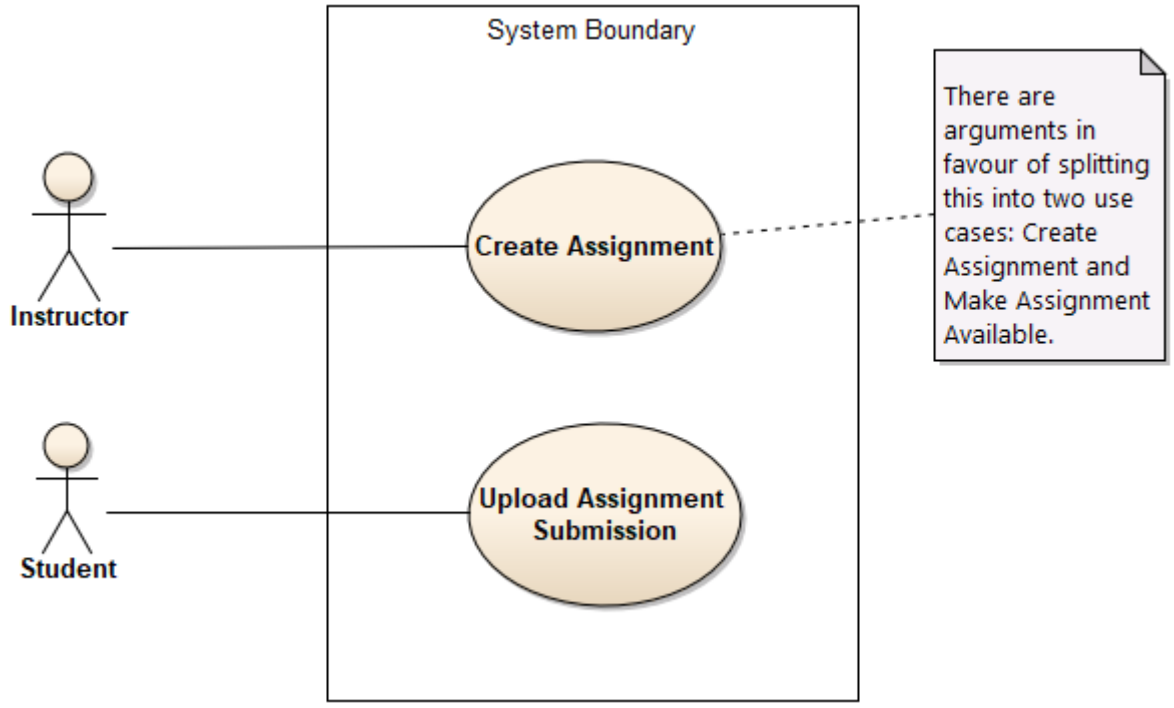

Figure 3: Use case diagram for Q1.2

Part 2 Dynamic Domain Modelling

Make sure your system sequence diagrams match your domain model. For example, you should use the same class and attributes names.

The sequence diagrams assume the users have been authenticated. A user does not typically login to perform a single activity, and for many system, all activities require authentication so it makes no sense to clutter the diagrams with login messages unless there is something special/unusual about them.

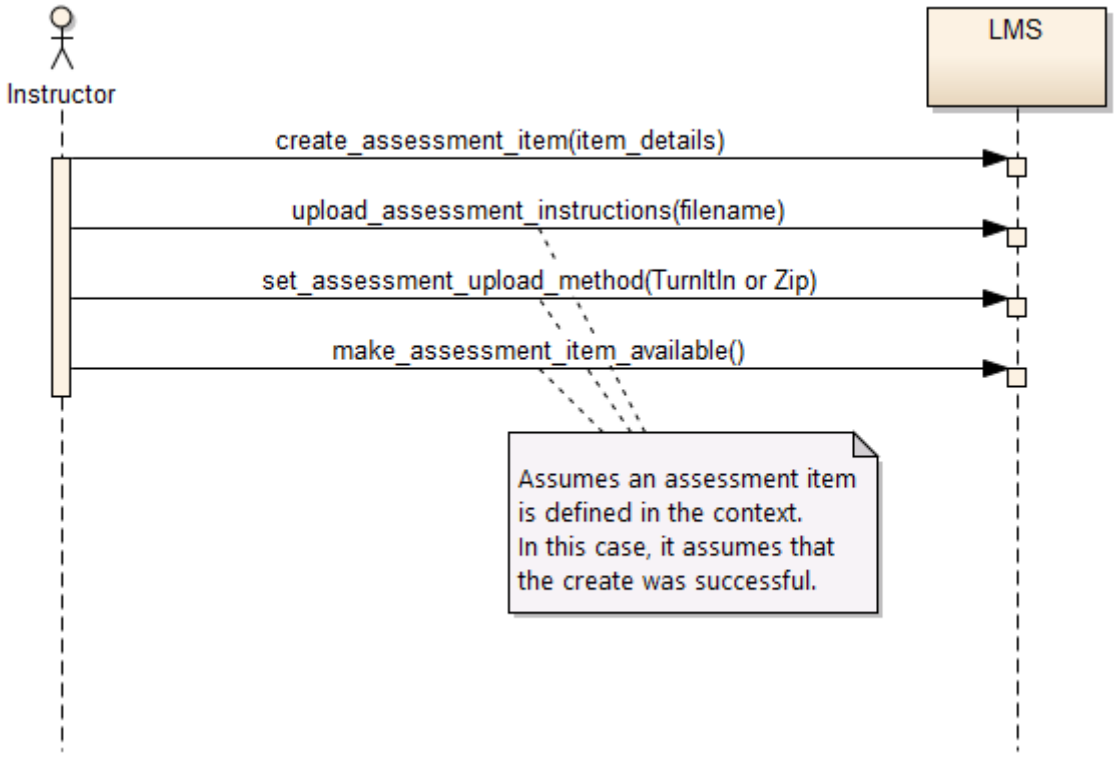

Figure 4: Create Assignment

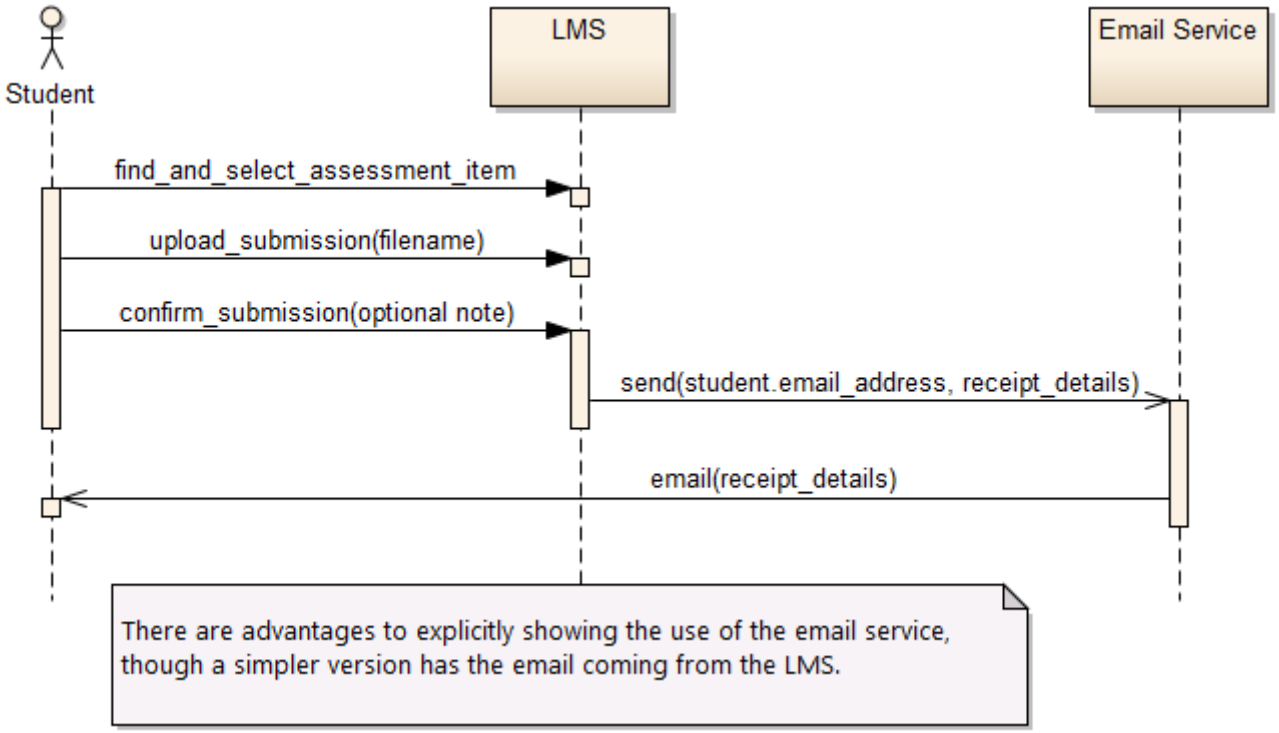

Figure 5: Upload Assignment Submission