1. Diagram

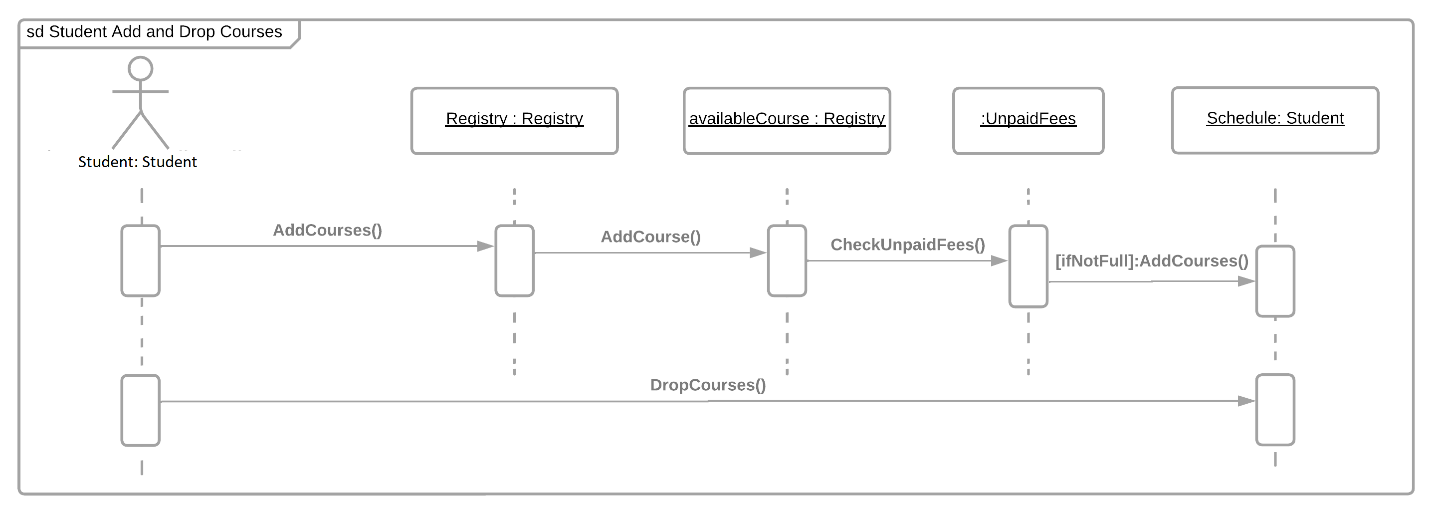
   Description automatically generatedDiagram

   Description automatically generated
2. aDiagram

   Description automatically generatedDiagram

   Description automatically generatedDiagram

   Description automatically generatedDiagram

   Description automatically generatedDiagram

   Description automatically generatedDiagram

   Description automatically generatedb Diagram

   Description automatically generatedDiagram

   Description automatically generatedDiagram

   Description automatically generatedDiagram

   Description automatically generatedDiagram

   Description automatically generatedDiagram

   Description automatically generatedDiagram

   Description automatically generatedc Diagram

   Description automatically generated
3. The Adapter Design Pattern is used to convert an interface of one class to provide functionality to another class that it would not have been able to. For instance, if there is a class that can only input specific results than another’s, the Adapter would be able to adapt/morph the input so that the other class can use that specific input. The example below shows how the Adapter converts variables meant for one object to another object. Text

   Description automatically generated

The Façade Design Pattern is used to create a new interface as a higher level of abstraction of one class so other classes can interact with it without accessing its internal components. For instance, if you had a couple of classes that you wanted to work with one another but did not want them to access all of the other classes’ internal attributes, a Façade would be able to represent some of those classes’ internal attributes as one class. Rather than molding a class to have it work with another class, you could create a class that has some of both classes’ attributes. Text

Description automatically generatedText

Description automatically generatedText

Description automatically generated