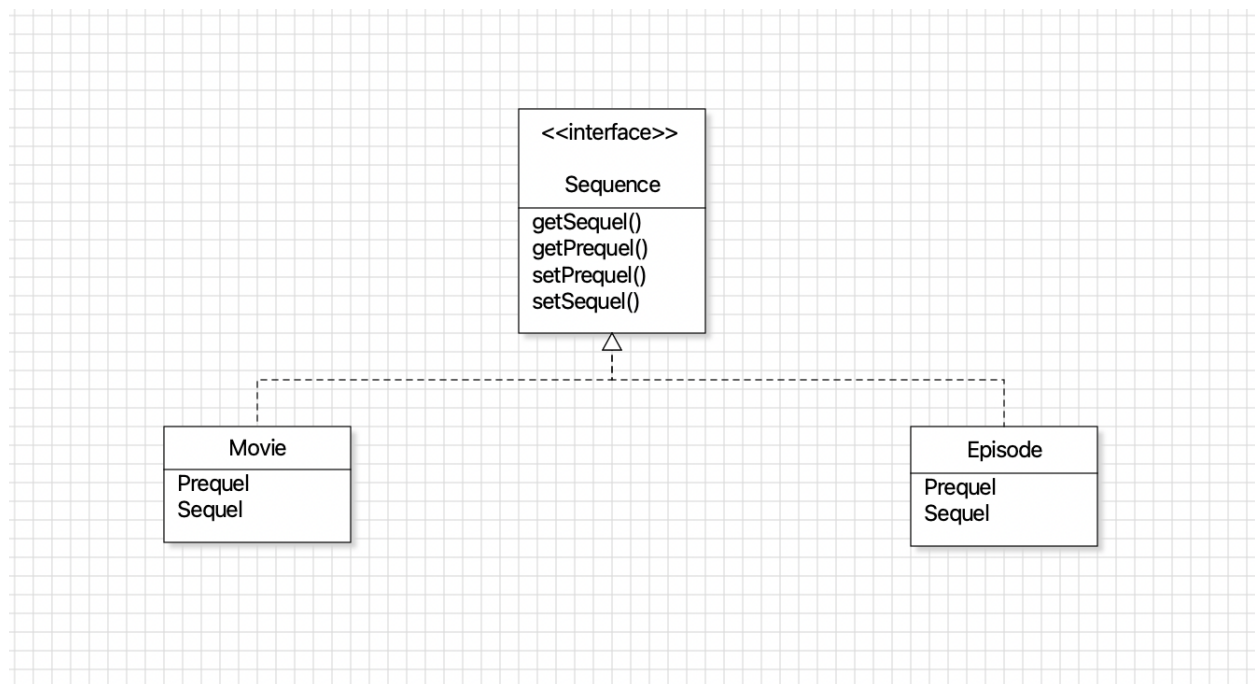


COMP303 Assignment 2, Name: Ryan Sowa, ID#: 260886668

In my design, I used the idea of the Interface Segregation Principle to write my interfaces so that the classes which implemented these interfaces would not have to implement any unused methods. I was forced, however, to use generics in some methods in my interfaces in order to accept arguments which could vary in type. The advantage of this is the diverse number of types a method could take; the disadvantage is that the generic type would have to be type-casted every time the method was implemented.

My “Sequence” interface was designed to be used to get and set preceding and following elements in Movie and Episode. Although this idea does simulate the idea of a sequence, not all movies and episodes have a “Prequel” and “Sequel” field. Please see the UML diagram below demonstrating this:



As a result, I had to constantly check in my methods if these private fields were null.