“An object model is a visual representation of a system’s objects, actions, and associated attributes.” (Adkisson, 2019). The visual nature can help to dispel any questions about the structure of the data as it is decomposed into classes. Moreover, the object model can show how classes are connected to the original object, i.e., that is graphically it would be easy to see what is a super or subclass.

The same way it is easy to see the structure of the super and subclasses any flavor of polymorphism can be seen quickly because the same method will be shown on each child class and maybe the parent class. This idea of polymorphism was shown in the textbook with the Undergraduate and Graduate students’ example both containing a calc-tuition method.

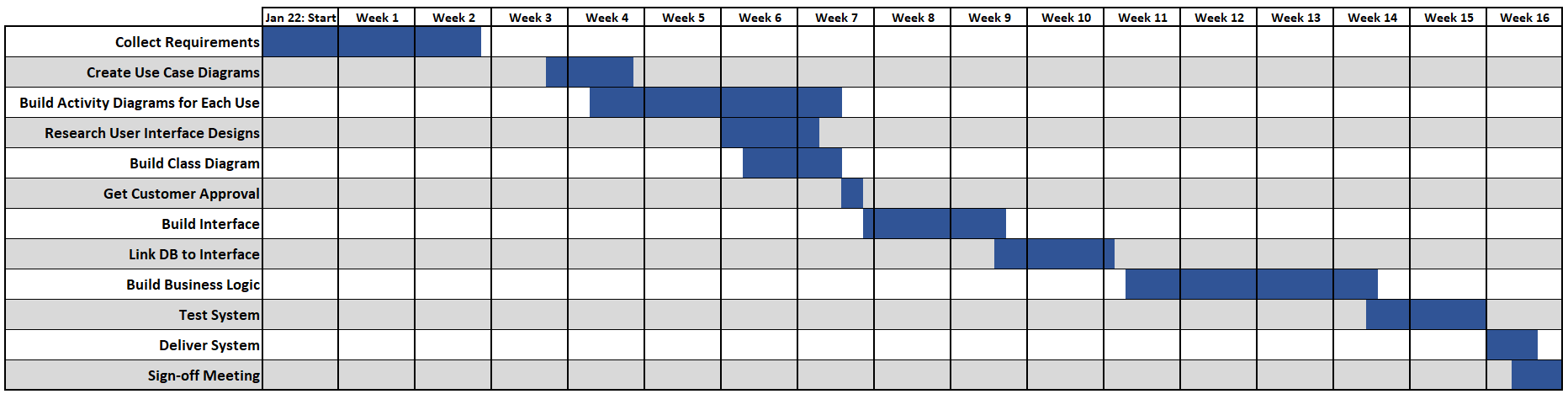
The idea of encapsulation is also represented visually by denoting private, public, and protected attributes and methods. By describing some attributes as private or protected the only way to manipulate the state of the attribute would be through the public methods, i.e., the getter and setter methods.

In looking at a system the behaviors or the object type and its associations may be diagramed. If wishing to capture the behavior of the system one would employ a process model; whereas an object model should be employed to map out the objects, their types, and any associations.

Just like an object or process model is a visual representation of the system a Gantt chart is a visual story of the timelines, responsibilities, and task dependencies of a project. They are useful for quickly comparing the current date and completeness versus the pre-computed expectations. What they are not good at showing is velocity – that is how quickly (or slowly) multiple tasks are completed. For example, if one were to look at 3 completed tasks, task one and two could have taken longer than expected but task two less time. A Gantt chart might show this as being on time (which it would be) but there would be no nuance within that data to suggest any inherent risks, e.g., maybe one engineer is slower than the others.

The Gantt charting tool in LucidChart was inefficient. There was no way to convert a date into a time block. To do so I would have had to use Excel to compute the timeline so I just made the chart within Excel also. Also, one of the key aspects of a Gantt chart, responsibility, was not clear from the interview transcript. A few of the tasks were assigned responsibilities but most were not so seeing if anyone was over taxed or underutilized was made difficult. Moreover, there was very little discussion of dependency.

Adkisson, H. (2019, March 30) *Object Modeling for Designers: An Introduction*. Medium. <https://hpadkisson.medium.com/object-modeling-for-designers-an-introduction-7871bdcf8baf>



Responses

1

Any thoughts on if you'd use a process or object model to map put behaviors or associations?

Also, I agree that the free Lucid Chart Gantt chart is not up to the full challenge. I ended up making one in Excel so that I can have full customization if I need it. I had challenges in assigning responsibilities as the interview does not go into details on every task - many are just mentioned with no work assignment.  
  
In the end, the shapes of our Gantt charts are very similar - both are able to display the vacation of one of the workers.