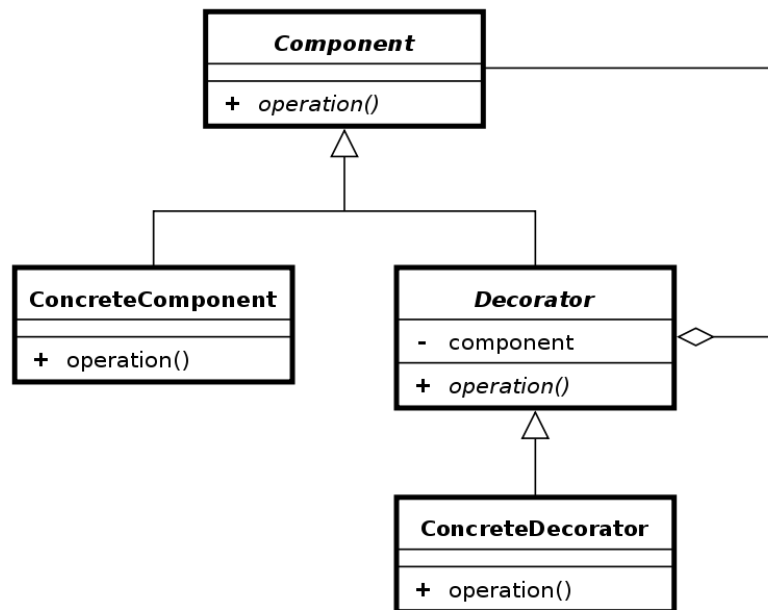


I use the **decorator design pattern** to achieve the design flexibility for dimensions D1-D3. The figure is list below:



I create one person interface, one concrete component called rider, one decorator called personDecorator and four concreteDecorator.

In order to achieve the goal of D1, I create one concreteDecorator called payment. In this class, the payment method is randomly arranged. In each concrete component or concreteDecorator, the board or unboard time is arranged differently. And the boarding and unboarding delay caused by speedup or multiple doors being used is set by a method called "speedup" in bus class (there I don't use bus decorator pattern because it is much simpler by using a single method).

The step is that the person factory creates normal person at first and inserts it into simulation. While during insert, the person's type changes from normal person to those with luggage and wheel-chair. Finally, different types of person are stored in the person list of simulation.