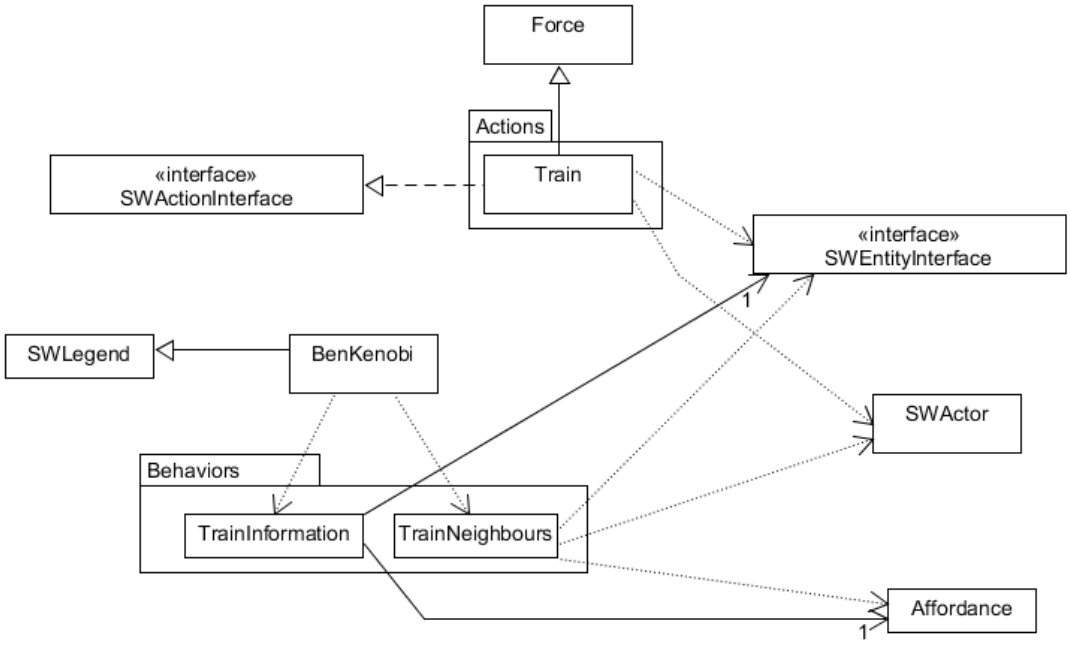

SWActionInterface - unmodified
Force - new
Train - new
SWActor - modified
SWEntityInterface – unmodified
SWLegend - unmodified
BenKenobi - modified
TrainInformation - new
TrainNeighbours - new
Affordance - unmodified

Design Doc

The functionality whereby Ben being able to trian Luke will be added as a type of behavior. Similar to how attack is implemented. The affordance Luke(Player) to be trainable will be added when the Actor is created. The current sequence of actions for Ben is the search for targets to attack then if there is he would attack them. If there was no targets to attack he would perform his patrol. TrainInformation and TrainNeighbours which work similarly to AttackInformation and AttackNeighbours will be added to his sequence. It will be added before patrol now making Ben's sequence try to attack, try to train, then patrol. Since force is added as an attribute similar to hitpoints, the increase of force level will be added as a method similar to takeDamage as well. Instead of decreasing hitpoints by damage, it would increase force level by an amount that is to be determined.

OnlyOkayDesigns also decided that Training actors would not be bound to Ben Kenobi only, several Actors(that do not exist as of now) can also train Actors that are trainable. The player

may also train Actors that are trainable if the player were to reach a high enough level for force. This would mean that the 'train' action would be an extension of the 'Force' class mentioned in ForceAbility_MindControl_DesignDoc.pdf. The condition of training remain as requiring a target that is trainable in the same space and having a high enough force level(for the player).

Design Rationale

Adding the training behavior to Ben Kenobi was the most straight forward way after analysing and observing the given implementation. Adding it before patrol and after attack makes it logical such that if enemies are present, Ben would attack them instead of training Luke in the presence of enemies. Having the train behavior work similarly to the attack behavior makes the code easier to understand .