Goonies, group 2

October 8, 2014

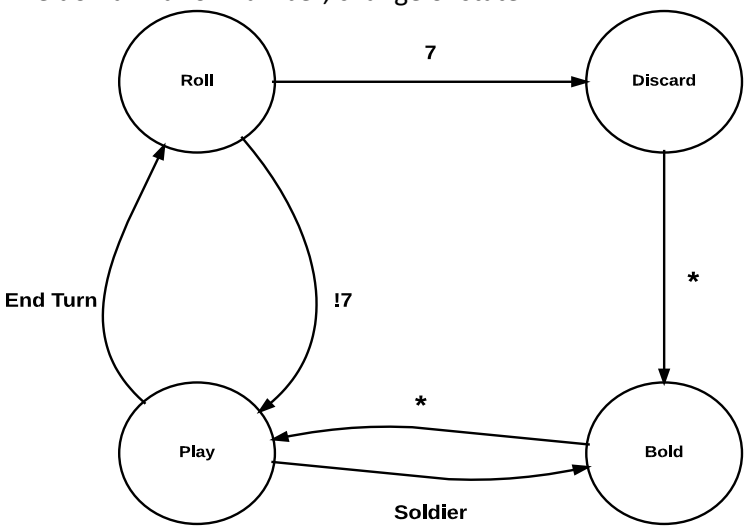
Phase 2 Design

1. Write a document explaining the following aspects of your design:
   a. **Explain in detail how you will implement the Observer pattern in your Model.**

      Within the client model there are classes that correspond to map views; each class corresponding to a map view will extend Observable while the corresponding controller will implement observer. After the player joins a game, the controllers will then register themselves with the observable. As the game progresses and the client model is updated, the client model checks the equivalence of its class fields with those of the incoming (updated) server model. If the class field is not equivalent, it (as an observable) will notify its observer (the corresponding controller), and in turn, the notified controller will then update its view.

   b. **Explain in detail how you will use the State pattern to change the behavior of your MapController depending on the current game state. What are the different states? What will be the MapController's behavior in each state?**
      i. The controller will have a state class, which allows certain functions depending on which of the four states it is set at.
      ii. Rolling - roll dice
      iii. Playing - building, buying, playing dev card, trading, finish turn
      iv. Robbing/Soldier - move robber, rob player
      v. Discarding - discard cards

      The domain: a roll number, change of state




2. See serverPollerSequenceUML.pdf, BuildRoadSequence.pdf