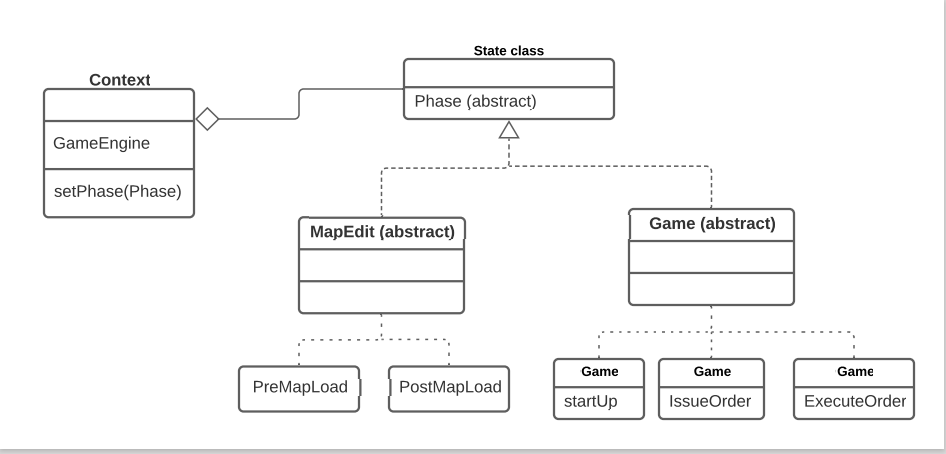
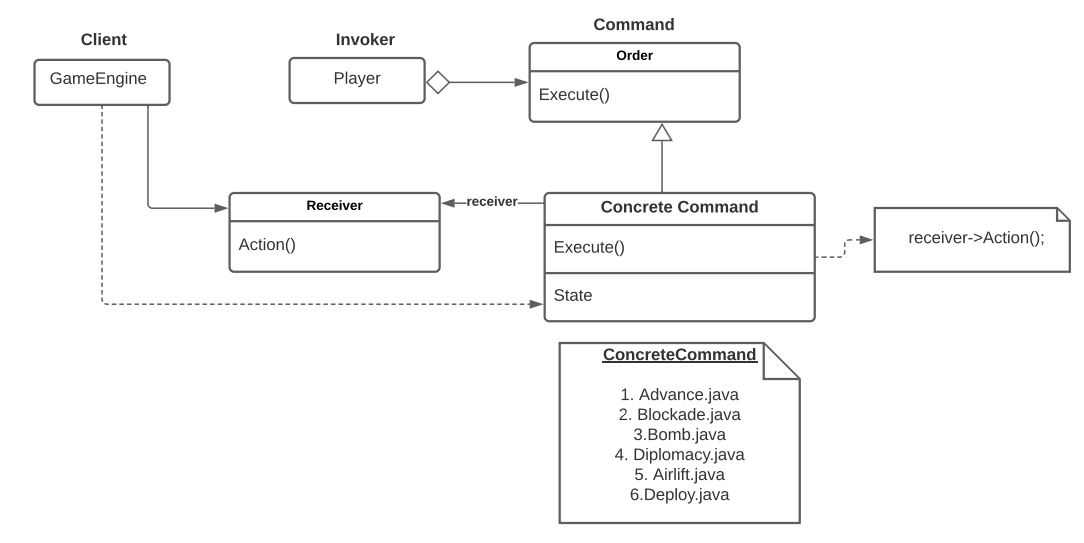
* In build 2 we have main **5 major changes** in our existing code which are as below: -

1. **Implementation of state pattern**
2. **Implementation of command pattern**
3. **Implementation of observer pattern**
4. **Implementation of different orders for player**
5. **Advance**
6. **Airlift**
7. **Bomb**
8. **Blockade**
9. **Diplomacy**
10. **Refactoring in exist code**
11. **State Pattern: -**



* Here context class is GameEngine class.
* State class is Phase class.
* To implement state pattern in our existing code we mainly divide our existing MapEdit and Game classes into different phases.
* MapEdit class further divided into below phases
  + PreMapLoad
  + PostMapLoad
* Game classes divided into below phases
* StartUp
* IssueOrder
* ExecuteOrder

1. **Command Pattern: -**



* Here client is GameEngine class.
* Invoker is Player class.
* Command is Order class.
* To implement command pattern in our existing code we mainly make 6 different concrete command for different orders the concrete command for command pattern as below
* Advance.java
* Blockade.java
* Bomb.java
* Diplomacy.java
* Airlift.java
* Deploy.java
* The orders are created as the player executes its issue\_order () method, and the orders are executed when the GameEngine gets the Player’s orders from the Players using the next\_order () method, then executes the orders by calling the execute () method of the Order.

1. **Refactoring: -**
2. Created new CreateMap class and refactored it from EditMap.
3. Created new LoadMap class and refactored it from EditMap.
4. Separated Random number generation logic from AssignCountries.
5. Used Random Number genaration code in AssignCard to to keep Code DRY (Don’t Repeat Yourself).
6. Refactored Random Number generation code to method overiding that can help to generate random number with upper bound considering lower bound to be zero or within upper bound and lower bound.
7. updated method signature of CreateMap to pass in map variable to keep the map model consistent and avoid creating new local copy.
8. Updated method signature of loadMap to pass in map varible to keep the map model consistent and avoid creating new local copy.
9. Updated the return type of createMap to return Map variable.
10. Updated the return type of loadMap to return Map variable.
11. Refactored Player model class to provide functionality with cards.
12. Created a common gameEngine from map engine and gameengine (removed old mapengine and gameengine) transfered control from main class to gameengine class for entire game State pattern for better state handling command pattern for commands moved game related functions to game utils added temp order inside player class to fetch order while issueing it without any arguments or return value old issue order method used to take order input from player class, removed it. deploy order function was inside player class, removed it
13. Removed deploy order logic from Player class
14. Implmetation of state pattern
15. Implementation of command pattern
16. Implementation of observer pattern