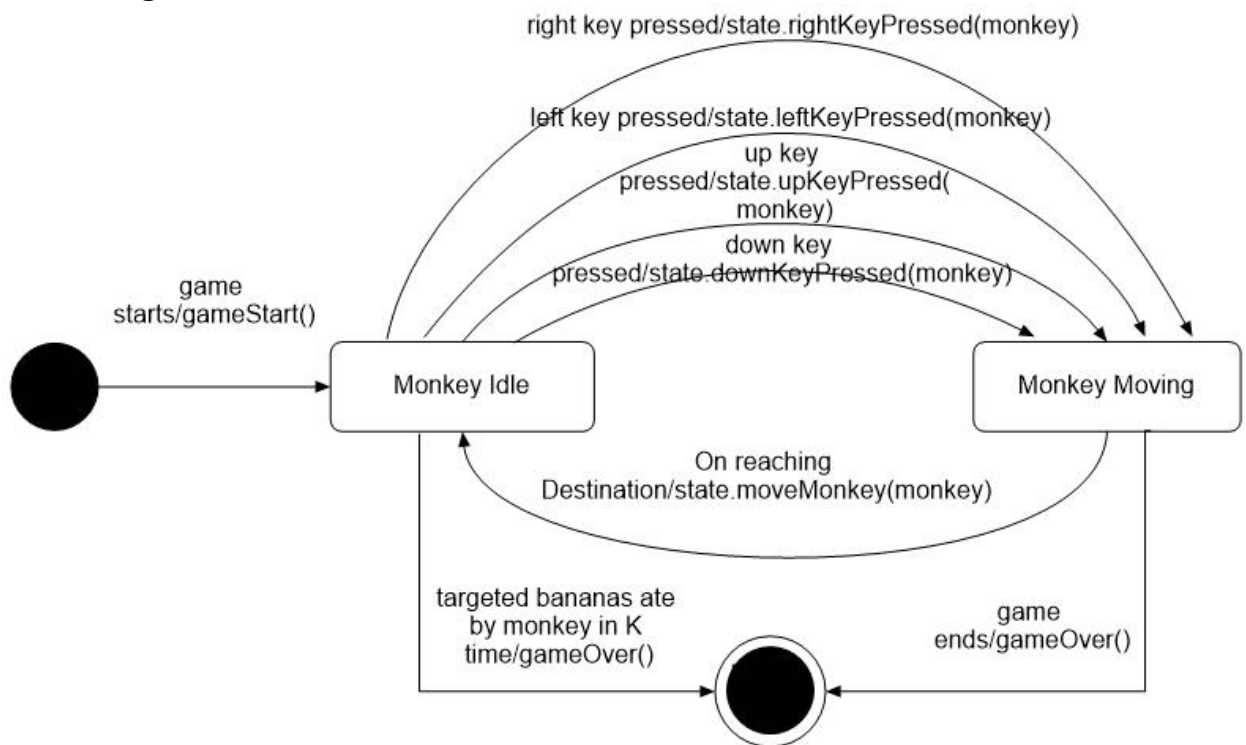


Monkey Banana Game

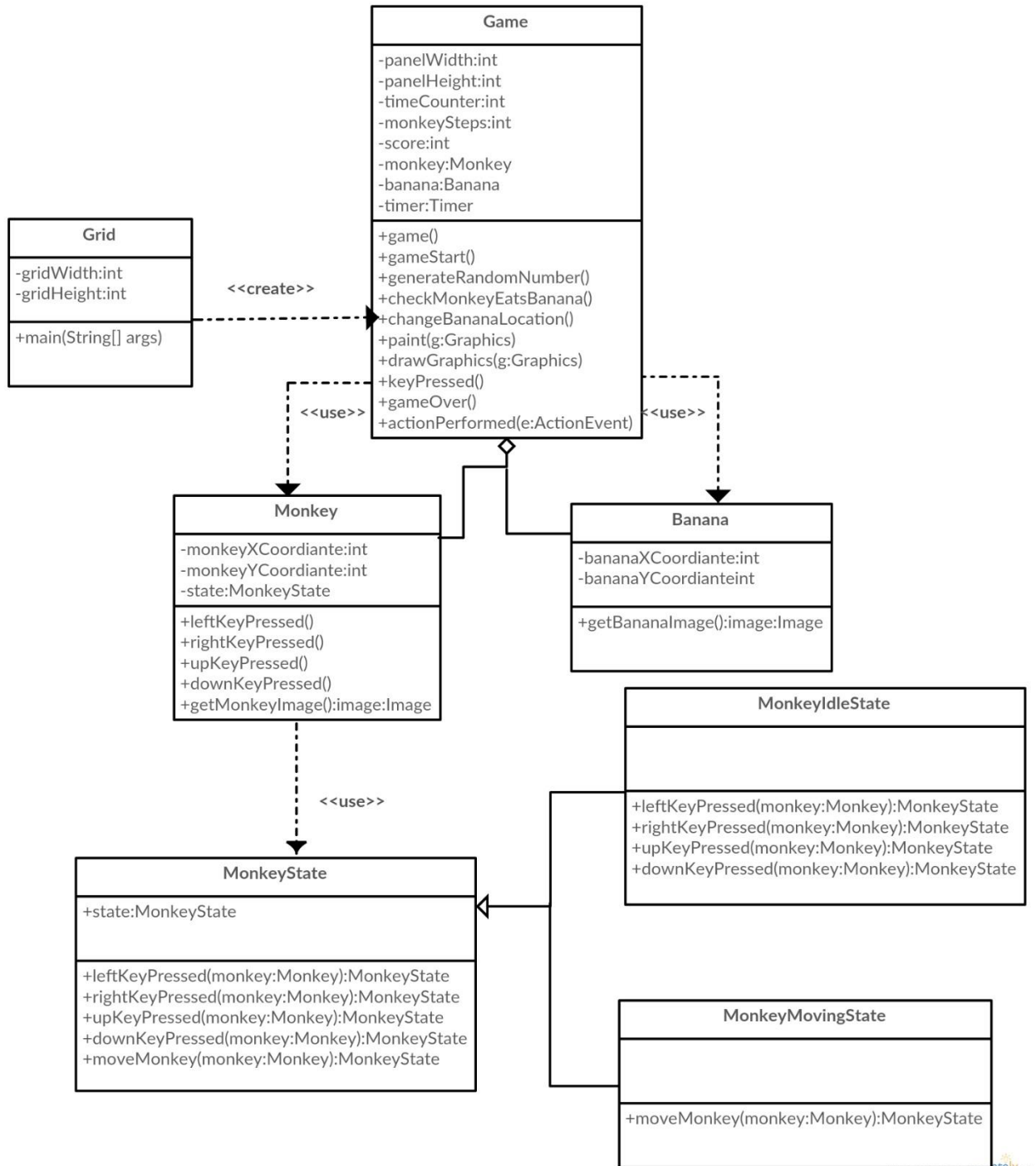
Design:

The game consist of 7 classes named Grid, Game, Monkey, Banana, MonkeyState, MonkeyIdleState and MonkeyMovingState. The Grid class creates the Game class i.e. it call the Game class constructor. The Game class uses the Monkey and Banana class. State design pattern is implemented with client as Game class, Monkey as subject, MonkeyState as the base class for the 2 states of the monkey represented as 2 classes MonkeyIdleState and MonkeyMovingState which are the subclasses of the MonkeyState class. When event occurs i.e. key pressed (left, right, up, down) the monkey moves from idle state to moving state as represented by classes MonkeyMovingState and MonkeyIdleState.

State Diagram



Design Class Diagram



Implementation:

The game timer starts on click of the start button with timer time of $K = 30000$ milliseconds i.e. 30 secs. The Monkey can move $N = 20$ steps left, right, top and down. The size of each step is 25 px. So the total panel size is $25 * 20 = 500 \times 500$. Each banana changes its location in every 7000 milliseconds i.e. 7 seconds. If the monkey eats banana a new banana timer is started for 7 secs. If the monkey eats 5 bananas (100 points for each banana) i.e. $5 \times 100 = 500$ points (score) in 30 secs then the game is over and the user wins the game else the user loses the game.

The game can be restarted again by clicking on the start button. Game is build using JFrame and JPanel. Game class extends JPanel and Grid class extends the JFrame.

Sample Code:

```
public static void main(String[] args){
    EventQueue.invokeLater(new Runnable() {
        public void run(){
            new Grid();
        }
    });
}

package monkey.banana.game;

public class MonkeyIdleState extends MonkeyState {

    public MonkeyState leftKeyPressed(Monkey monkey){
        if(monkey.getXCoordinate() > 0 &&
Game.gameTimer.isRunning()){
            monkey.setXCoordinate(monkey.getXCoordinate() -
Monkey.gridSquare);
            return new MonkeyMovingState(monkey);
        }
        return this;
    }

    public MonkeyState rightKeyPressed(Monkey monkey){
        if(monkey.getXCoordinate() < 475 &&
Game.gameTimer.isRunning()){
```

```

        monkey.setXCoordinate(monkey.getXCoordinate()+Monkey.gridSquare);
        return new MonkeyMovingState(monkey);
    }
    return this;
}

    public MonkeyState upKeyPressed(Monkey monkey) {
        if(monkey.getYCoordinate() > 0 &&
Game.gameTimer.isRunning()) {
            monkey.setYCoordinate(monkey.getYCoordinate()-
Monkey.gridSquare);
            return new MonkeyMovingState(monkey);
        }
        return this;
    }

    public MonkeyState downKeyPressed(Monkey monkey) {
        if(monkey.getYCoordinate() < 475 &
Game.gameTimer.isRunning()) {

            monkey.setYCoordinate(monkey.getYCoordinate()+Monkey.gridSquare);
            return new MonkeyMovingState(monkey);
        }
        return this;
    }
}

```

Source Code

MonkeyState.java

```

package monkey.banana.game;

public class MonkeyState {
    MonkeyState state;

    public MonkeyState leftKeyPressed(Monkey monkey) {
        return state;
    }

    public MonkeyState rightKeyPressed(Monkey monkey) {
        return state;
    }

    public MonkeyState upKeyPressed(Monkey monkey) {
        return state;
    }
}

```

```

    public MonkeyState downKeyPressed(Monkey monkey) {
        return state;
    }

    public MonkeyState moveMonkey(Monkey monkey) {
        return state;
    }
}

```

MonkeyIdleIdleState.java

```

package monkey.banana.game;

public class MonkeyIdleState extends MonkeyState {

    public MonkeyState leftKeyPressed(Monkey monkey) {
        if(monkey.getXCoordinate() > 0 && Game.gameTimer.isRunning()){
            monkey.setXCoordinate(monkey.getXCoordinate() -
Monkey.gridSquare);
            return new MonkeyMovingState(monkey);
        }
        return this;
    }

    public MonkeyState rightKeyPressed(Monkey monkey) {
        if(monkey.getXCoordinate() < 475 && Game.gameTimer.isRunning()){

monkey.setXCoordinate(monkey.getXCoordinate()+Monkey.gridSquare);
            return new MonkeyMovingState(monkey);
        }
        return this;
    }

    public MonkeyState upKeyPressed(Monkey monkey) {
        if(monkey.getYCoordinate() > 0 && Game.gameTimer.isRunning()){
            monkey.setYCoordinate(monkey.getYCoordinate() -
Monkey.gridSquare);
            return new MonkeyMovingState(monkey);
        }
        return this;
    }

    public MonkeyState downKeyPressed(Monkey monkey) {
        if(monkey.getYCoordinate() < 475 & Game.gameTimer.isRunning()){

monkey.setYCoordinate(monkey.getYCoordinate()+Monkey.gridSquare);
            return new MonkeyMovingState(monkey);
        }
        return this;
    }
}

```

MonkeyMovingState.java

```
package monkey.banana.game;

public class MonkeyMovingState extends MonkeyState {

    public MonkeyMovingState(Monkey monkey) {
        moveMonkey(monkey);
    }

    public MonkeyState moveMonkey(Monkey monkey) {
        try {
            Thread.sleep(50);
        } catch (InterruptedException e) {
            e.printStackTrace();
        }
        return new MonkeyIdleState();
    }

}
```