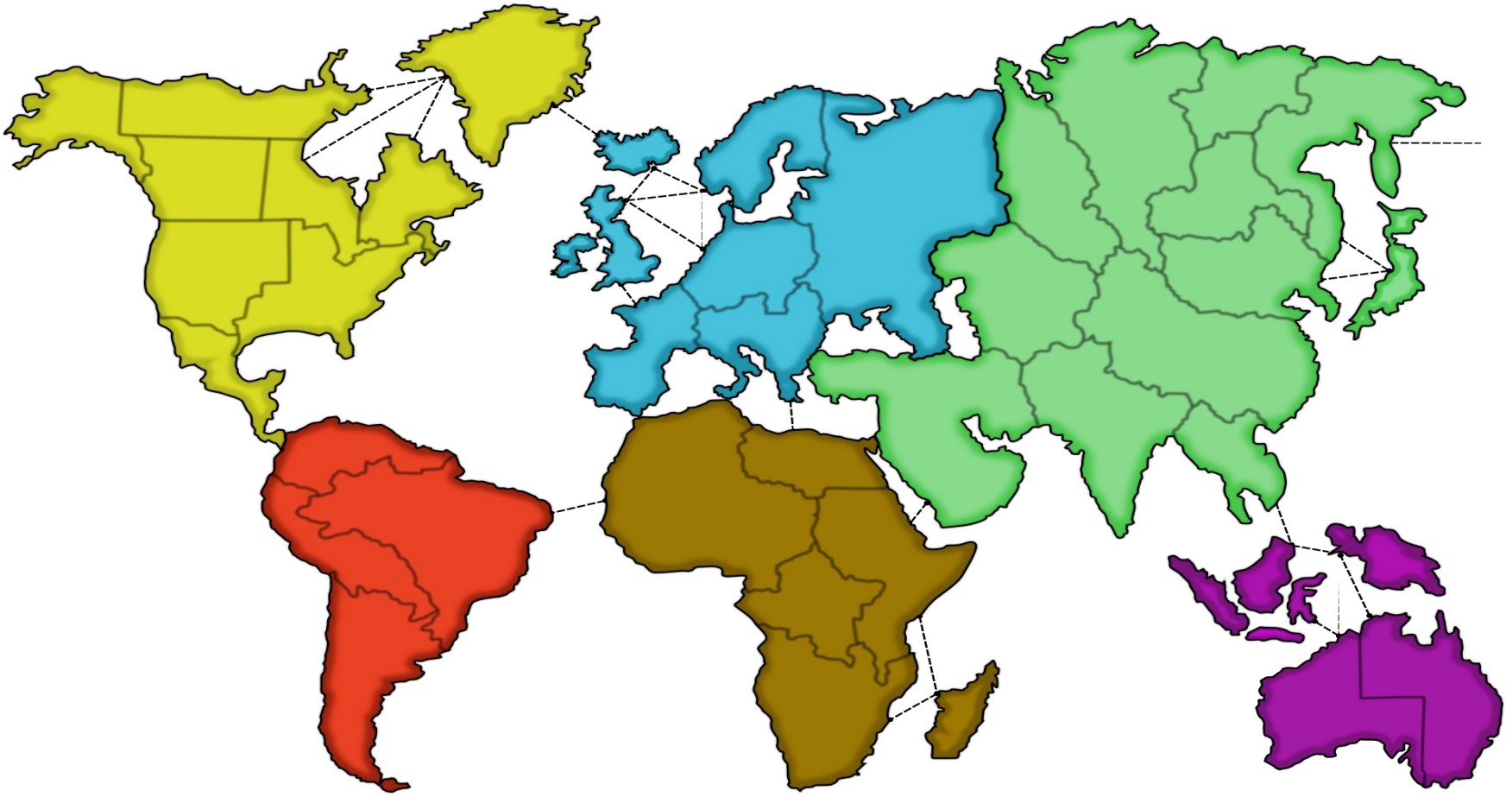
Group 3

yeet.



CreatePlan

****

1. _____
2. _____
3. _____

1. Play when # of players ≥ 3
2. Stop when # of players < 3

Order

1. Color
2. Turn Order
3. Number of Armies

Turns start

Get new armies and assign them to territories

Attack

1. Dice
2. Territories attackable

Win

Move armies into acquired territories.

Lose

At minimum.

Fortify (think about specifics)

Check and repeat...

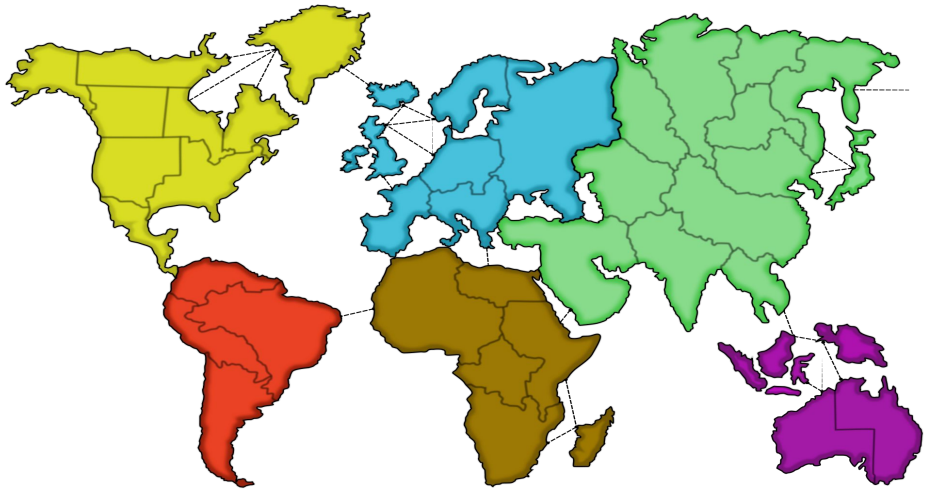
End game

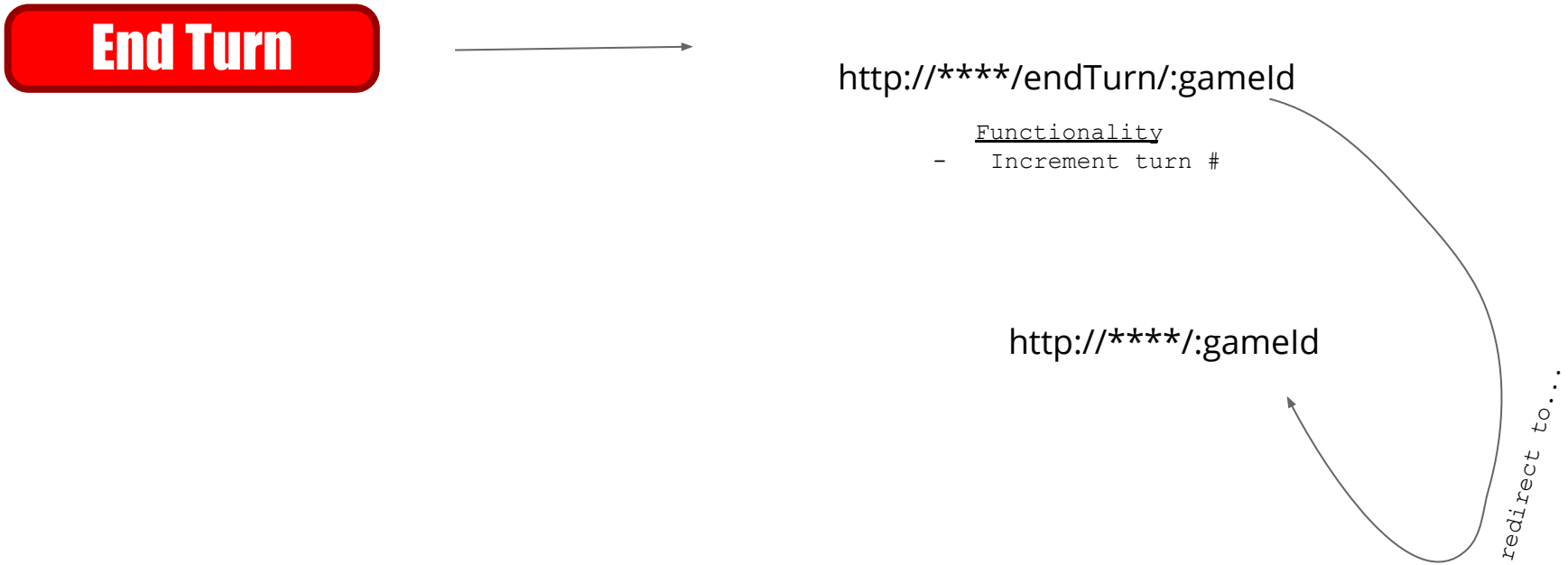
```
Turn Number: 0 //so Rahul2's turn
Players Turn Order:
1.   Rahul2
2.   Rahul1
3.   Rahul3

Countries:
1.   Country1: 1, Rahul2
2.   Country2: 3, Rahul3
3.   Country3: 4, Rahul1
4.   Country4
5.   Country5
6.   Country6: 3, Rahul1
```

End Turn

Calculate current player turn using turn number modulo number of players -> update number of placeable armies -> on click, decrement player placeable armies, increment territory armies -> on 0 placeable armies -> POST request to end turn -> turn number incremented -> repeat loop




End Turn
http://****/endTurn/:gameId
Functionality
- Increment turn #
http://****/:gameId
redirect to...

Game State

http://****/territoryInfo/:territoryId/:gameId

http://****/gameInfo/:gameId

http://****/territoriesInfo/:gameId

http://****/playerInfo/:playerOrder/:gameId

http://****/playersInfo/:gameId

Game Functionality

- http://****/addArmiesToTerritory/:amount/:territoryId/:gameId
  - Add armies to territory

- Disable elements when they should be disabled
-

New Turn

Turn # Changes

Immediate

- Find who new player is
- Reward armies
- Go into assigning state

http://****/addArmiesToTerritory/:amount/:territoryId/:gameId

Assigning

Remaining armies: 4

Click on territory to add to it!

End Turn

Fortify Phase

Attack Phase

Territory Class

var armies: Int

var owner: Player

var adjacents: List[Territory]

def startPlay()

If game isn’t won

- Iterate through player list and calculate new armies per player turn
- Increment turn counter for each player turn