**Documentation**

**The Node Class**

Attributes:

1. Id - special identifier represented by a number
2. nodeNeighbours – a list that contains the neighborhood nodes
3. HasRoadWith –a list that contains the nodes which the node has roads with
4. hexNeighbours – a list with the neighborhood hexes
5. hasPlayer – a bool variable which specifies if a node has a settlement from a player
6. playerSettled – the player which has the settlement on the node
7. hasSettlement - a bool variable which tells us if it has a settlement or not
8. settlementType – an object of type settlement which memorizes the type of settlement
9. hasPort - a bool variable which tells us if the settlement has a port or not;
10. portTradedResource – the resources specific to the port
11. portType – memorizes the type of the port

Methods:

1. Node( int id) – the constructor which initialize the id
2. ToString – a function that outputs all the data about the node
3. Getters and Setters

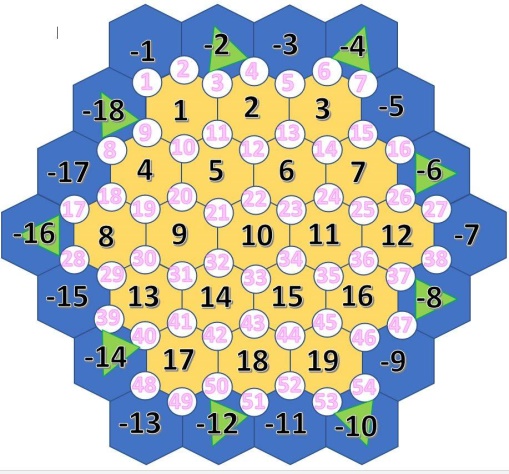
**Hex Class**

Attributes:

1. id – special identifier for each hex element
2. number – the number on the top of each hexagon
3. hasRobber – bool variable that specifies if a robber is placed on the hex
4. nodeNeighbors – list that contains all the node neighbors of the hex
5. hexNeigbords – list that contains all the hex neighbors
6. resource – contains the resource specifics

Methods:

1. Hex (int id, Resources resource, int number) – initializes the id, resource, number, and hasRobber with value false
2. Hex (int id) – initializes the id of the hex
3. setDetails – sets the resource and number to the given parameters
4. toString - a function that outputs all the data about the hex
5. Getters and Setters

**Map Class**

Attributes:

1. nodes - the list of nodes that belong to the map;
2. hexes - the list of hexes that belongs to the map;
3. resourceTile – a list that contains the resources found on the map;
4. typeOfPort - a list that contains the types of ports.

Methods:

1. map – constructor that initializes the map and outputs the map;
2. initNodes – initializes 54 nodes in the nodes list;
3. initHexes – initializes 37 hexes on the in the hexes list(19 of them with positive values and the rest of them with negative values, for the ones on the edge of the map so we can build settlements on the edge of the map)
4. generateHexDetails – assigns resources for every land tile except the desert and randomly puts the tiles on the map; the ocean tiles are assigned ocean resources
5. addPortsToNodes – generates ports on the map; a normal port will follow a specific resource port and vice versa; the specific resource ports will be randomized;
6. addHexNeighboursToHexes, addNodeNeighboursToHexes, addHexNeighboursToNodes, addNoteNeighbourstoNodes - hardcoded initialization of the map
7. Getters and Setters.

**Player Class**

Attributes:

1. Name - name of the player
2. id - Id of the player
3. points - the number of points that the player accumulated
4. allRoads - the list of roads that the player built
5. settledNodes - the list of settlements that the player built
6. wheatQty,sheepQty,clayQty,stoneQty,woodQty - the number of resources cards of each type that the player has
7. villagesLeft,citiesLeft,roadsLeft - the number of settlements and roads the player can still build
8. knightCardsLeft,victoryPointCardsLeft,roadBuildingCardsLeft,yearOfPlentyCardsLeft,monopolyCardsLeft - the number of development cards the player has
9. soldierCardsUsed - number of solider cards used by the player
10. hasLongestRoad - bool variable that specifies if the player has the achievement for the longest road;

Methods:

1. Player() - constructor that initializes all the attributes that belong to the player
2. ToString() - outputs the player name
3. useKnightCard(),useRoadBuildingCard(),useYearOfPlentyCard(),useMonopolyCard() – functions that implement the usage of the development cards;
4. Getters and Setters

**Gamestate Class**

Attributes:

1. Map - an object of type map which ………
2. Players - a list that contains the players which currently play the game
3. knightCardsLeft,victoryPointCardsLeft,roadBuildingCardsLeft,yearOfPlentyCardsLeft,monopolyCardsLeft - variables which contain the number of development cards of each type left;

Methods:

1. GameState() - constructor which initializes the player list on the map and the number of development cards left;
2. Getters and Setters

**Road Class**

Attributes:

1. node1,node2 - objects of type Node which belong to the road;
2. owner – a Player type object that contains the player that owns the road

Methods:

1. Setters and Getters

**Ports Class -** class which contains an enumeration of types of ports (simple and resource-specific)

**Program Class -** Main class of the project

**Resources Class -** class which contains an enumeration of types of resources that the tiles contain (including dessert and ocean types of resources)

**Settlement Class** - class which contains an enumeration of types of settlements (village or cities);

**Shuffler Class** -Class that contains the shuffling method which puts the elements of a list on new, randomized positions .