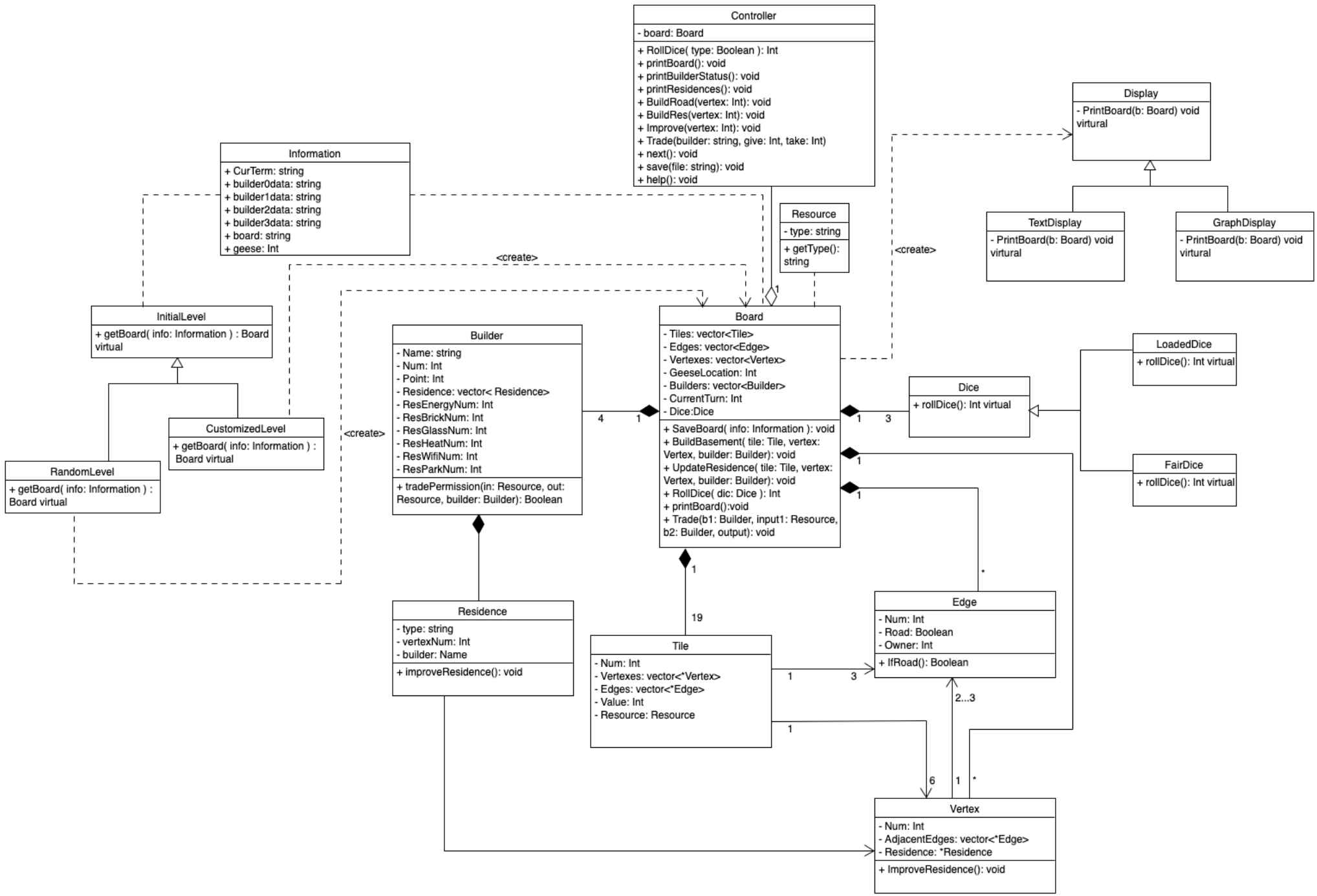
Controller
- board: Board
+ RollDice( type: Boolean ): Int
+ printBoard(): void
+ printBuilderStatus(): void
+ printResidences(): void
+ BuildRoad(vertex: Int): void
+ BuildRes(vertex: Int): void
+ Improve(vertex: Int): void
+ Trade(builder: string, give: Int, take: Int)
+ next(): void
+ save(file: string): void
+ help(): void
Display
- PrintBoard(b: Board) void virtural
Information
+ CurTerm: string
+ builder0data: string
+ builder1data: string
+ builder2data: string
+ builder3data: string
+ board: string
+ geese: Int
Resource
- type: string
+ getType(): string
TextDisplay
- PrintBoard(b: Board) void virtural
GraphDisplay
- PrintBoard(b: Board) void virtural
<create>
<create>
<create>
1
InitialLevel
+ getBoard( info: Information ) : Board virtual
Board
- Tiles: vector<Tile>
- Edges: vector<Edge>
- Vertexes: vector<Vertex>
- GeeseLocation: Int
- Builders: vector<Builder>
- CurrentTurn: Int
- Dice:Dice
+ SaveBoard( info: Information ): void
+ BuildBasement( tile: Tile, vertex: Vertex, builder: Builder): void
+ UpdateResidence( tile: Tile, vertex: Vertex, builder: Builder): void
+ RollDice( dic: Dice ): Int
+ printBoard():void
+ Trade(b1: Builder, input1: Resource, b2: Builder, output): void
Builder
- Name: string
- Num: Int
- Point: Int
- Residence: vector< Residence>
- ResEnergyNum: Int
- ResBrickNum: Int
- ResGlassNum: Int
- ResHeatNum: Int
- ResWifiNum: Int
- ResParkNum: Int
+ tradePermission(in: Resource, out: Resource, builder: Builder): Boolean
LoadedDice
+ rollDice(): Int virtual
Dice
+ rollDice(): Int virtual
CustomizedLevel
+ getBoard( info: Information ) : Board virtual
RandomLevel
+ getBoard( info: Information ) : Board virtual
FairDice
+ rollDice(): Int virtual
4
1
1
3
1
1
1
*
Residence
- type: string
- vertexNum: Int
- builder: Name
+ improveResidence(): void
19
Tile
- Num: Int
- Vertexes: vector<*Vertex>
- Edges: vector<*Edge>
- Value: Int
- Resource: Resource
Edge
- Num: Int
- Road: Boolean
- Owner: Int
+ IfRoad(): Boolean
1
3
2...3
1
6
1
*
Vertex
- Num: Int
- AdjacentEdges: vector<*Edge>
- Residence: *Residence
+ ImproveResidence(): void