Node

+ position: Vector3

+ vertexIndex : int

+ Node(Vector2 : Node)

Square

+ topLeft, topRight, bottomRight, bottomLeft : ControlNode

+ centreTop, centreRight, centreBottom, centreLeft : Node

+ configuration: int

+ Square (ControlNode \_topLeft, ControlNode \_topRight, ControlNode \_bottomRight, ControlNode \_bottomLeft)

Control Node

+ active: bool

+ above, right : Node

+ ControlNode (Vector3 \_pos, bool \_active, float squareSize) : base(\_pos)

Mesh Generator

+walls: MeshFilter

+cave: MeshFilter

+ GenerateMesh(int[,] : map, float : squareSize

- CreateWallMesh( )

- CalculateMeshOutlines()

Room : IComparable<Room>

+ tiles : List<Coord>

+ edgeTiles : List<Coord>

+ connectedRooms : List<Room>

+ roomSize : int

+ isAccessibleFromMainRoom : bool

+ isMainRoom : bool

+ Room(List<Coord> roomTiles, int[,] map)

+SetAccessibleFromMainRoom

+ConnectedRooms (Room roomA, Room roomB)

Map Generator

+ seed: string

+ useRandomSeed : bool

+ randomFillPercent : int

- map : int[,]

- GenerateMap( )

- ProcessMap( )

- RandomFillMap( )

- List<List<Coord>> GetRegions(int tileType)

- SmoothMap( )

- Coord : struct