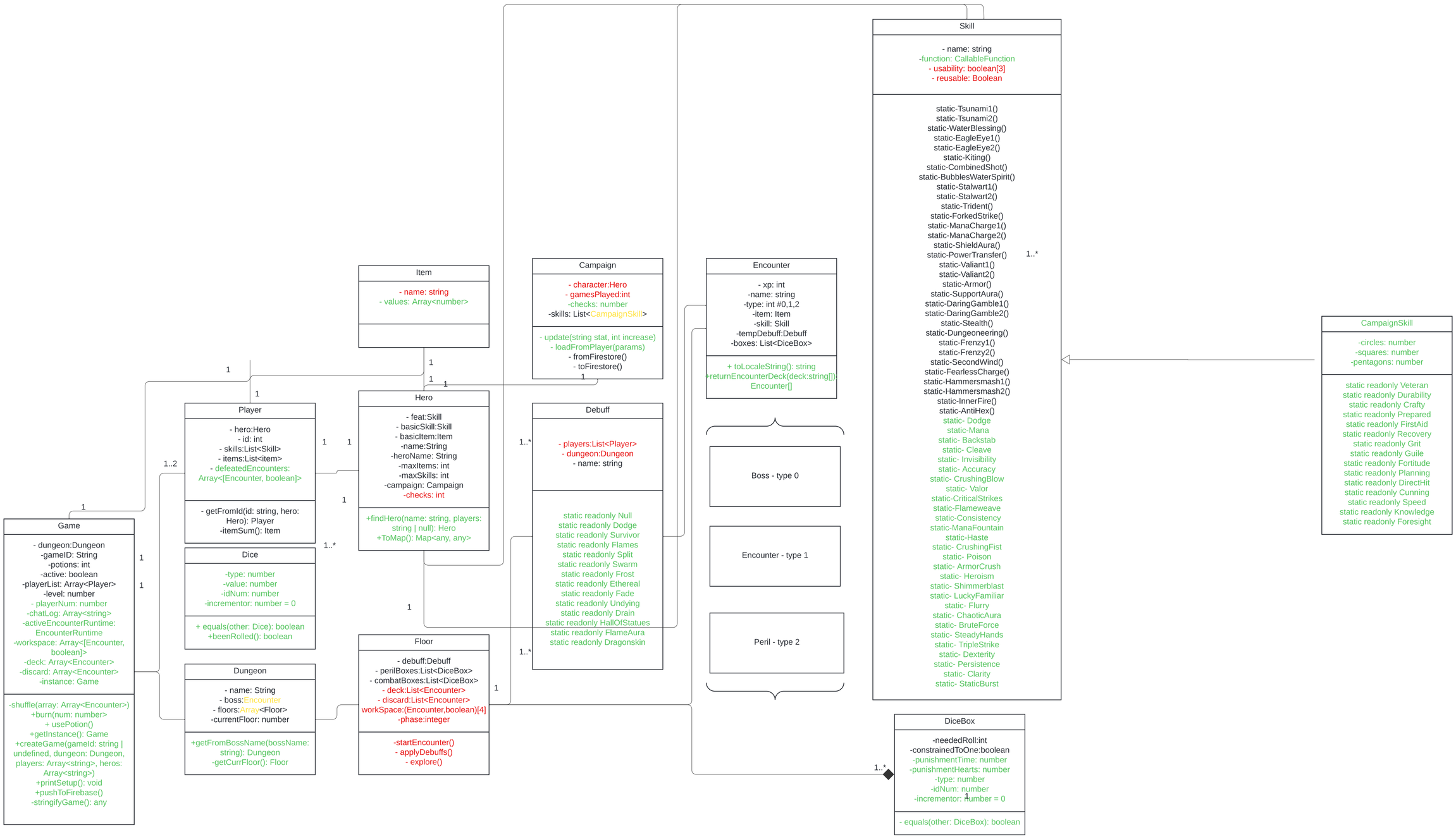
Skill
- name: string
-function: CallableFunction
- usability: boolean[3]
- reusable: Boolean
static-Tsunami1()
static-Tsunami2()
static-WaterBlessing()
static-EagleEye1()
static-EagleEye2()
static-Kiting()
static-CombinedShot()
static-BubblesWaterSpirit()
static-Stalwart1()
static-Stalwart2()
static-Trident()
static-ForkedStrike()
static-ManaCharge1()
static-ManaCharge2()
static-ShieldAura()
static-PowerTransfer()
static-Valiant1()
static-Valiant2()
static-Armor()
static-SupportAura()
static-DaringGamble1()
static-DaringGamble2()
static-Stealth()
static-Dungeoneering()
static-Frenzy1()
static-Frenzy2()
static-SecondWind()
static-FearlessCharge()
static-Hammersmash1()
static-Hammersmash2()
static-InnerFire()
static-AntiHex()
static- Dodge
static-Mana
static- Backstab
static- Cleave
static- Invisibility
static- Accuracy
static- CrushingBlow
static- Valor
static-CriticalStrikes
static-Flameweave
static-Consistency
static-ManaFountain
static-Haste
static- CrushingFist
static- Poison
static- ArmorCrush
static- Heroism
static- Shimmerblast
static- LuckyFamiliar
static- Flurry
static- ChaoticAura
static- BruteForce
static- SteadyHands
static- TripleStrike
static- Dexterity
static- Persistence
static- Clarity
static- StaticBurst
1..*
Item
- name: string
- values: Array<number>
Campaign
- character:Hero
- gamesPlayed:int
-checks: number
-skills: List<CampaignSkill>
- update(string stat, int increase)
- loadFromPlayer(params)
- fromFirestore()
- toFirestore()
Encounter
- xp: int
-name: string
-type: int #0,1,2
-item: Item
-skill: Skill
-tempDebuff:Debuff
-boxes: List<DiceBox>
+ toLocaleString(): string
+returnEncounterDeck(deck:string[]): Encounter[]
Boss - type 0
Encounter - type 1
Peril - type 2
CampaignSkill
-circles: number
-squares: number
-pentagons: number
static readonly Veteran
static readonly Durability
static readonly Crafty
static readonly Prepared
static readonly FirstAid
static readonly Recovery
static readonly Grit
static readonly Guile
static readonly Fortitude
static readonly Planning
static readonly DirectHit
static readonly Cunning
static readonly Speed
static readonly Knowledge
static readonly Foresight
Player
- hero:Hero
- id: int
- skills:List<Skill>
- items:List<item>
- defeatedEncounters: Array<[Encounter, boolean]>
- getFromId(id: string, hero: Hero): Player
-itemSum(): Item
Hero
- feat:Skill
- basicSkill:Skill
- basicItem:Item
-name:String
-heroName: String
-maxItems: int
-maxSkills: int
-campaign: Campaign
-checks: int
+findHero(name: string, players: string | null): Hero
+ToMap(): Map<any, any>
Debuff
- players:List<Player>
- dungeon:Dungeon
- name: string
static readonly Null
static readonly Dodge
static readonly Survivor
static readonly Flames
static readonly Split
static readonly Swarm
static readonly Frost
static readonly Ethereal
static readonly Fade
static readonly Undying
static readonly Drain
static readonly HallOfStatues
static readonly FlameAura
static readonly Dragonskin
Game
- dungeon:Dungeon
-gameID: String
-potions: int
-active: boolean
-playerList: Array<Player>
-level: number
- playerNum: number
-chatLog: Array<string>
-activeEncounterRuntime: EncounterRuntime
-workspace: Array<[Encounter, boolean]>
-deck: Array<Encounter>
-discard: Array<Encounter>
-instance: Game
-shuffle(array: Array<Encounter>)
+burn(num: number>
+ usePotion()
+getInstance(): Game
+createGame(gameId: string | undefined, dungeon: Dungeon, players: Array<string>, heros: Array<string>)
+printSetup(): void
+pushToFirebase()
-stringifyGame(): any
Dice
-type: number
-value: number
-idNum: number
-incrementor: number = 0
+ equals(other: Dice): boolean
+beenRolled(): boolean
Dungeon
- name: String
- boss:Encounter
- floors:Array<Floor>
-currentFloor: number
+getFromBossName(bossName: string): Dungeon
-getCurrFloor(): Floor
Floor
- debuff:Debuff
- perilBoxes:List<DiceBox>
- combatBoxes:List<DiceBox>
- deck:List<Encounter>
- discard:List<Encounter>
workSpace:(Encounter,boolean)[4]
-phase:integer
-startEncounter()
- applyDebuffs()
- explore()
DiceBox
-neededRoll:int
-constrainedToOne:boolean
-punishmentTime: number
-punishmentHearts: number
-type: number
-idNum: number
-incrementor: number = 0
- equals(other: DiceBox): boolean
1
1..2
1..*