

GameController
+ totalEnemies: int + enemyPrefab: GameObject + spawnedEnemies: List<GameObject>
+ Start(): void + Update(): void + SpawnEnemies(): void

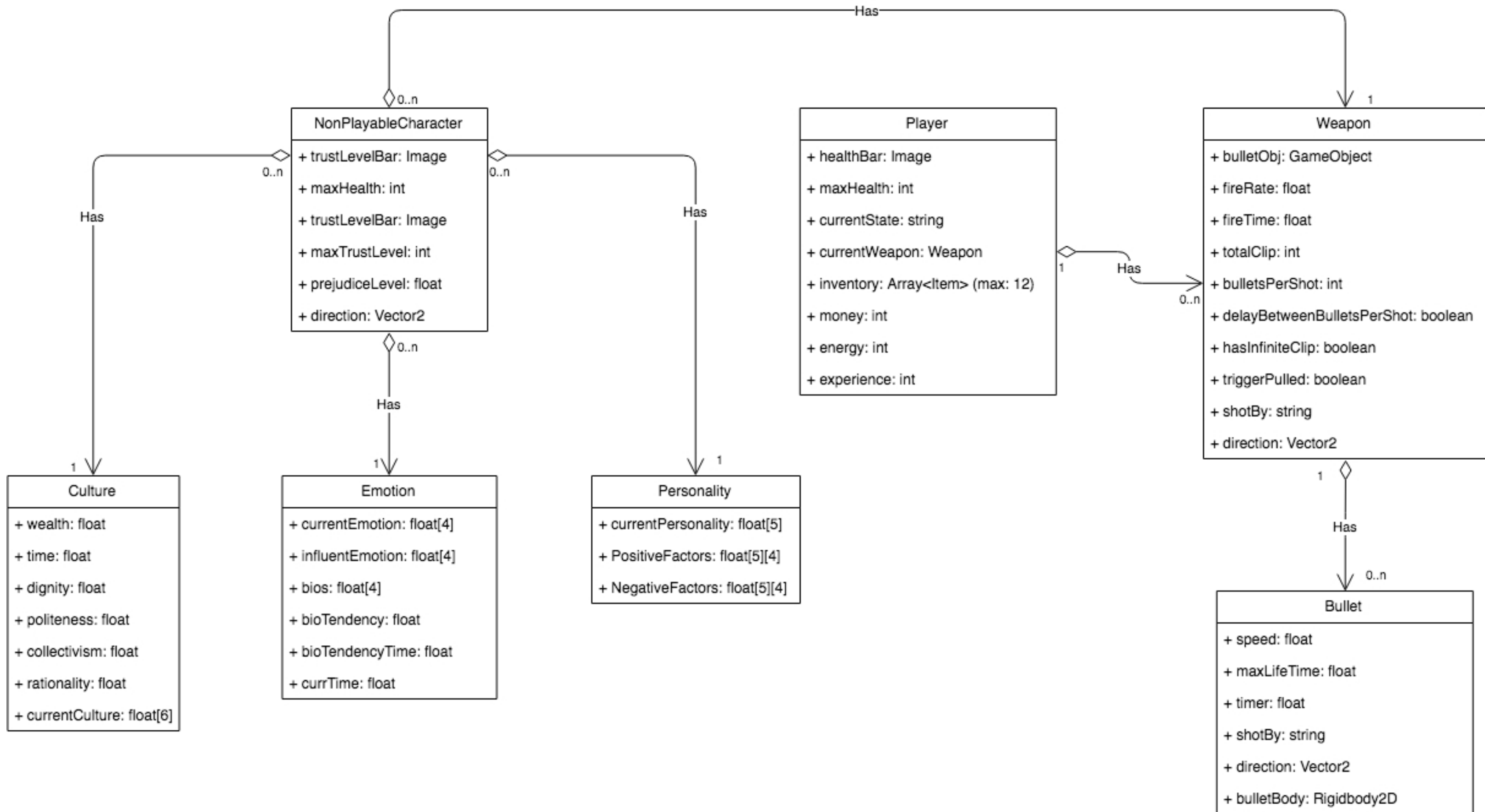
EnemyController
+ same class attributes
+ Start(): void + Update(): void + GenerateInitialEmotion(): void + GenerateInitialPersonality(): void + GenerateInitialCulture(): void + SetRandomDirection(): void + UpdateNormalMovement(float dt): void + UpdateStopAction(float dt): void + UpdateBehavior(float dt): void + UpdateTrustLevel(): void + DispatchPlayerState(string playerState): void + UpdateEmotionByEvent(float[] eventEmotion): void

PlayerController
+ same class attributes
+ Start(): void + Update(): void + UpdateNormalMovement(float dt): void + OnTriggerEnter2D(Collider2D collision): void + OnDrawGizmosSelected(): void

ProxemicsController
+ relatedEnemy: Enemy + proxemicValues: float[] + proxemicStates: string[] + maxProxemicsTimer: float + proxemicsTimer: float + currentPlayer: Player + otherEnemies: List<GameObject>
+ Start(): void + Update(): void + OnTriggerEnter2D(Collider2D collision): void + OnTriggerExit2D(Collider2D collision): void + OnDrawGizmosSelected(): void + SetCurrentPlayer(PlayerController pObj): void

BulletController
+ same class attributes
+ Start(): void + Update(): void + SetDirection(float dirX, float dirY): void + SetShotBy(string someone): void + GetShotBy(): string

WeaponController
+ same class attributes
+ Start(): void + Update(): void + SetDirection(float dirX, float dirY): void + SetShotBy(string someone): void + GetShotBy(): string + SetTrigger(boolean value): void + Shoot(): void



CultureController

+ same class attributes

+ Culture(float[] newCulture): void
+ InitializeCulture(float[] newCulture): void
+ LoadCultureDict(Dictionary<string, float> dict)
+ GetCulture(): float[]
+ GetTime(): float
+ GetWealth(): float
+ GetDignity(): float
+ GetPoliteness(): float
+ GetRationality(): float
+ GetCollectivism(): float

PersonalityController

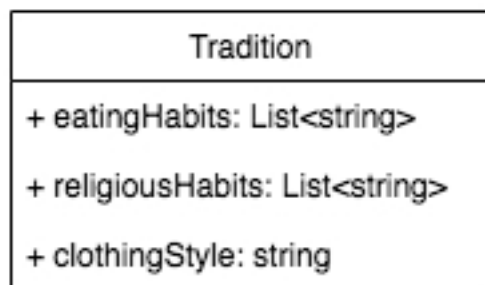
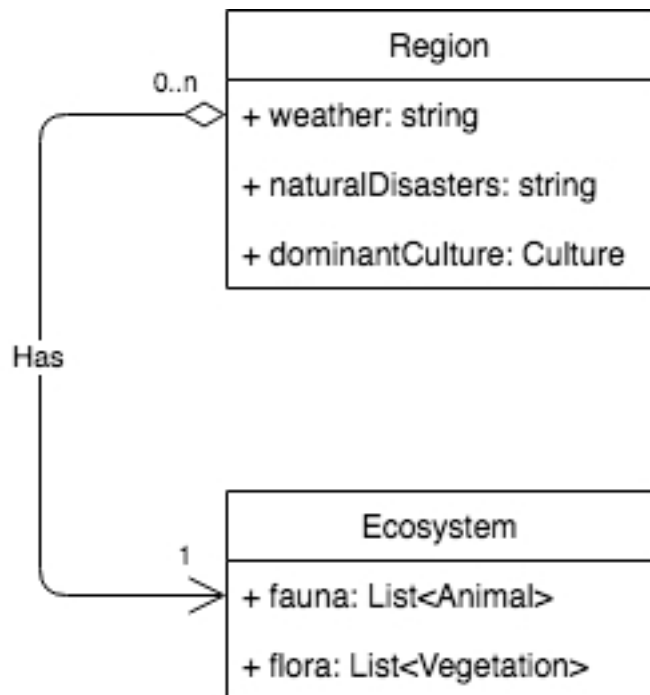
+ same class attributes

+ Personality(float[] newPersonality): void
+ GetPersonality(): float[]
+ InitializePersonality(float[] newPersonality): void

EmotionController

+ same class attributes

+ Emotion(float[] newEmotion)
+ GetEmotion(): float[]
+ GetMostInfluentEmotion(): float[]
+ LoadBios(): void
+ InitializeEmotion(float[] newEmotion): void
+ AddEmotion(float[] otherEmotion): void
+ IsBios(): boolean
+ Update(float dt): void
+ GetName(): string
+ GetMostInfluentName(): string
+ GetMentalStateName(): string
+ GetRandomEmotion(): float[]
+ SetMostInfluentEmotion(): void
+ ClampCurrentEmotion(): void
+ ResetCurrentEmotion(): void



PlayerController
+ energyBar: Image + lifeBar: Image + moneyText: Text
+ Start(): void + Update(): void + InputHandler(): void + OnCollisionEnter(Collider): void + GetTotalMoney(): float + SetTotalMoney(float): void

MenuController
+ pauseMenu: GameObject + gameOverMenu: GameObject + inGameMenu: GameObject
+ Start(): void + Update(): void + LoadLevel(string): void

CharacterController
+ currentEmotion: Emotion + currentState: SituationGraph + character: Character
+ Start(): void + Update(): void + ChangeEmotion(Emotion, Personality): void + TakeDecision(Action): void

RegionController
+ currentRegion: GameObject + isDungeon: bool
+ Start(): void + Update(): void + GenerateMap(): void + SpawnItems(): void + SpawnEnemies(): void + CreatePuzzle(): void