**UIApplicationDelegate:** *protocol*
----------------
----------------

**UIResponder:** *class*
----------------
----------------

*conforms to*

*inherits*

**AppDelegate:** *class*
----------------
- var window: UIWindow?
----------------
- application(_ application: UIApplication, didFinishLaunchingWithOptions launchOptions: [UIApplication.LaunchOptionsKey: Any]?) -> Bool
- applicationWillResignActive(_ application: UIApplication)
- applicationDidEnterBackground(_ application: UIApplication)
- applicationWillEnterForeground(_ application: UIApplication)
- applicationDidBecomeActive(_ application: UIApplication)

**UIViewController:** *class*
----------------
----------------

**SKPhysicsContactDelegate:** *protocol*
----------------
----------------

*inherits*

**GameViewController:** *class*
----------------
- var shouldAutorotate: Bool
- var supportedInterfaceOrientations: UIInterfaceOrientationMask
- var prefersStatusBarHidden: Bool
----------------
- viewDidLoad()

*conforms to*

**PhysicsCategories:** *enum*
----------------
- _let_teamCategory:UInt32_
----------------

**TeamCharacteristic:** *enum*
----------------
- _let_defaultSize:CGFloat_
----------------

**ZPosition:** *enum*
----------------
- _let_background:CGFloat_
- _let_team:CGFloat_
- _let_label:CGFloat_
----------------

**GameStatus:** *enum*
----------------
----------------

**Effects:** *enum*
----------------
----------------

**PlayerError:** *enum*
----------------
----------------

*inherits*

**Error:** *class*
----------------
----------------

**Team:** *class*
----------------
- var name : String
- var colour : UIColor
- var radius : Int
- var status : Bool
- var imageName: String
- var sprite: SKSpriteNode?
- var players : [Player]
- var checkedPlayers : [Player]
- _let MAX_NUMBER_OF_PLAYER_ : Int_
- _let MIN_NUMBER_OF_PLAYER_ : Int_
- _let BASE_RADIUS_ : Int_
----------------
- getName() -> String
- setName(newName : String)
- getColour() -> UIColor
- setColour(newColour : UIColor)
- getRadius() -> Int
- getImageName() -> String
- setImageName(newImageName : String)
- getSprite() -> SKSpriteNode
- getPlayers() -> [Player]
- setPlayers(newPlayers : [Player])
- showPlayers() -> [Player]
- generatePlayers()
- isEqual(_ team: Team) -> Bool
- calculateRadius()
- init(newName:String, newColour:UIColor, newImageName:String)

**Player:** *class*
----------------
- _var HEALTH_INCREASE_BY_
- var health: Int
- var name : String
- var selectedWeapon : Int
- _var_MAX_HEALTH_ : Int_
- var alive : Bool
- var weaponsDelegate:[WeaponDelegate]
----------------
- isAlive() -> Bool
- setAlive(alive : Bool)
- getName() -> String
- getWeapon() -> [WeaponDelegate]
- getSelectedWeapon() -> Int
- addWeapon(weapon : WeaponDelegate)
- setSelectedWeapon(selectedWeapon : Int)
- setHealth(health : Int) throws
- getHealth() -> Int
- getSelectedWeaponDamage() -> Int
- increaseHealth()
- attack(otherPlayer : Player)
- upgradeWeapon()
- dodge() -> Bool
- init(pName : String , pHealth : Int )

**ScoreScene:** *class*
----------------
----------------
- didMove(to view: SKView)
- touchesBegan(_ touches: Set<UITouch>, with event: UIEvent?)
- showLabels()

**MenuScene:** *class*
----------------
----------------
- didMove(to view: SKView)
- addLogo()
- addLabels()
- animate(label: SKLabelNode)
- touchesBegan(_ touches: Set<UITouch>, with event: UIEvent?)

*inherits*

**SKScene:** *class*
----------------
----------------

*inherits*

**undefined:** *extension*
----------------
----------------
- didBegin(_ contact: SKPhysicsContact)

**Conflict:** *class*
----------------
----------------
- fight(_ team1: inout Team, _ team2: inout Team) -> Team
- init()

**UMP:** *class*
----------------
- let damageMultiplier: Float
----------------
- fire() -> Int
- init(numBullets: Int, name: String)

**M249:** *class*
----------------
- let damageMultiplier: Float
----------------
- fire() -> Int
- init(numBullets: Int, name: String)

**Weapon:** *class*
----------------
- var name: String
- var numBullets: Int
- var damage :Int
- let maxNoOfBullets: Int
- _let DEFAULT_DAMAGE_
----------------
- getName() -> String
- setName(name: String)
- getNumBullets() -> Int
- setNumBullets(numBullets :Int)
- getDamage() -> Int
- setDamage(damage :Int)
- reload()
- fire() -> Int
- init(_ numBullets :Int, _ name: String, _ newDamage:Int)

**NightHawk:** *class*
----------------
- let damageMultiplier: Float
----------------
- fire() -> Int
- init(numBullets: Int, name: String)

**Maverick:** *class*
----------------
- let damageMultiplier: Float
----------------
- fire() -> Int
- init(numBullets: Int, name: String)

**GameScene:** *class*
----------------
- let bitMarkTeams:Int32
- var teams:[Team]
- var teamLeft:Int
- var status
----------------
- setupPhysics()
- didMove(to view: SKView)
- layoutScene()
- addTeamLabel(team:Team, position:CGPoint)
- touchesBegan(_ touches: Set<UITouch>, with event: UIEvent?)
- play()
- start()
- generateRandomLocation() -> (CGFloat, CGFloat)
- moveSprites()
- stopSprites()
- flagEffect(_ flagName:String) -> [SKTexture]
- effect(_ type:Effects) -> [SKTexture]
- addBackground()
- showTeams()
- getTeamByName(teamName:String)->Team
- getIndexOfTeamByName(teamName:String)->Int
- showWinner(team:Team)
- animateFlagWinner(teamName:String)
- updateLabel(team:Team)
- updateSize(team:Team)
- destroyTeam(team:Team)
- playSound(soundVariable : SKAction)
- saveScore(team:Team)

*inherits*

**Bullup:** *class*
----------------
- let damageMultiplier: Float
----------------
- fire() -> Int
- init(numBullets: Int, name: String)

**WeaponDelegate:** *protocol*
----------------
----------------
- getName() -> String
- setName(name: String)
- getNumBullets() -> Int
- setNumBullets(numBullets :Int)
- getDamage() -> Int
- setDamage(damage :Int)

*inherits*

setDamage(damage :Int)
- reload() -> Void
- fire() -> Int

- checkWinner() -> Bool