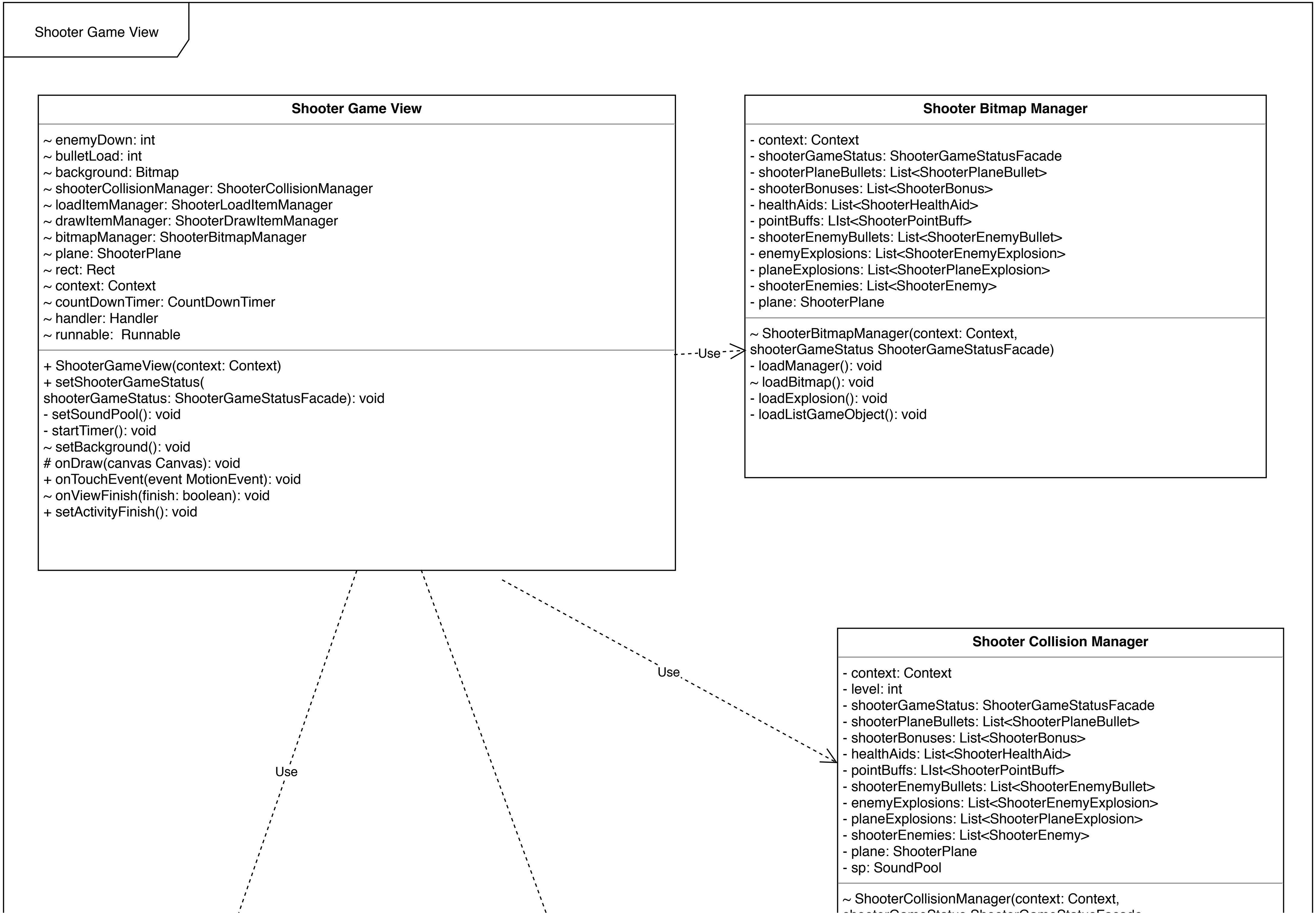


Shooter Game View

~ enemyDown: int
~ bulletLoad: int
~ background: Bitmap
~ shooterCollisionManager: ShooterCollisionManager
~ loadItemManager: ShooterLoadItemManager
~ drawItemManager: ShooterDrawItemManager
~ bitmapManager: ShooterBitmapManager
~ plane: ShooterPlane
~ rect: Rect
~ context: Context
~ countDownTimer: CountDownTimer
~ handler: Handler
~ runnable: Runnable

+ ShooterGameView(context: Context)
+ setShooterGameStatus(
shooterGameStatus: ShooterGameStatusFacade): void
- setSoundPool(): void
- startTimer(): void
~ setBackground(): void
onDraw(canvas Canvas): void
+ onTouchEvent(event MotionEvent): void
~ onViewFinish(finish: boolean): void
+ setActivityFinish(): void

Shooter Bitmap Manager

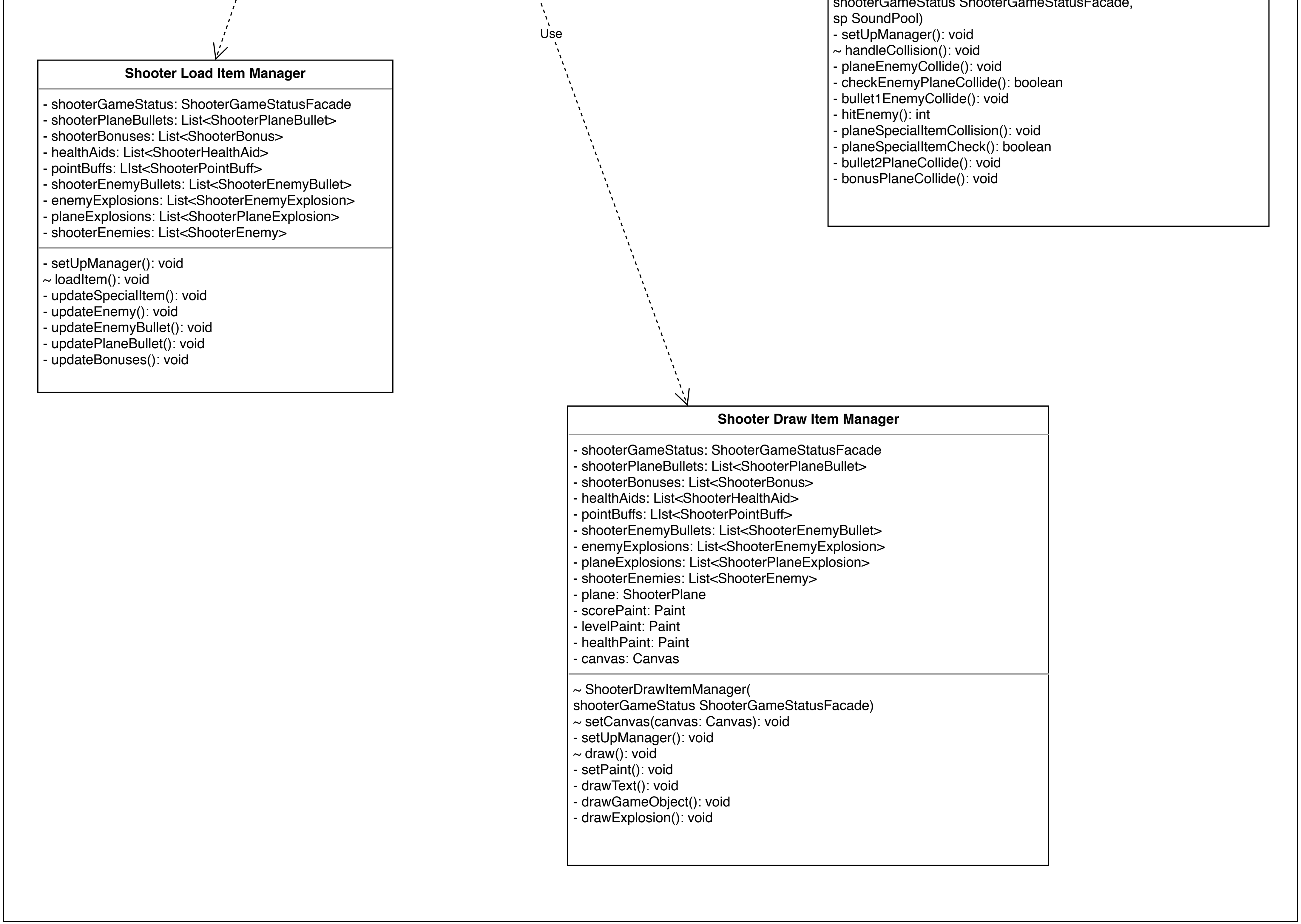
- context: Context
- shooterGameStatus: ShooterGameStatusFacade
- shooterPlaneBullets: List<ShooterPlaneBullet>
- shooterBonuses: List<ShooterBonus>
- healthAids: List<ShooterHealthAid>
- pointBuffs: LIst<ShooterPointBuff>
- shooterEnemyBullets: List<ShooterEnemyBullet>
- enemyExplosions: List<ShooterEnemyExplosion>
- planeExplosions: List<ShooterPlaneExplosion>
- shooterEnemies: List<ShooterEnemy>
- plane: ShooterPlane

~ ShooterBitmapManager(context: Context,
shooterGameStatus ShooterGameStatusFacade)
- loadManager(): void
~ loadBitmap(): void
- loadExplosion(): void
- loadListGameObject(): void

Shooter Collision Manager

- context: Context
- level: int
- shooterGameStatus: ShooterGameStatusFacade
- shooterPlaneBullets: List<ShooterPlaneBullet>
- shooterBonuses: List<ShooterBonus>
- healthAids: List<ShooterHealthAid>
- pointBuffs: LIst<ShooterPointBuff>
- shooterEnemyBullets: List<ShooterEnemyBullet>
- enemyExplosions: List<ShooterEnemyExplosion>
- planeExplosions: List<ShooterPlaneExplosion>
- shooterEnemies: List<ShooterEnemy>
- plane: ShooterPlane
- sp: SoundPool

~ ShooterCollisionManager(context: Context,
shooterGameStatus ShooterGameStatusFacade



shooterGameStatus ShooterGameStatusFacade,
sp SoundPool)
- setUpManager(): void
~ handleCollision(): void
- planeEnemyCollide(): void
- checkEnemyPlaneCollide(): boolean
- bullet1EnemyCollide(): void
- hitEnemy(): int
- planeSpecialItemCollision(): void
- planeSpecialItemCheck(): boolean
- bullet2PlaneCollide(): void
- bonusPlaneCollide(): void

Shooter Load Item Manager

- shooterGameStatus: ShooterGameStatusFacade
- shooterPlaneBullets: List<ShooterPlaneBullet>
- shooterBonuses: List<ShooterBonus>
- healthAids: List<ShooterHealthAid>
- pointBuffs: LIst<ShooterPointBuff>
- shooterEnemyBullets: List<ShooterEnemyBullet>
- enemyExplosions: List<ShooterEnemyExplosion>
- planeExplosions: List<ShooterPlaneExplosion>
- shooterEnemies: List<ShooterEnemy>

- setUpManager(): void
~ loadItem(): void
- updateSpecialItem(): void
- updateEnemy(): void
- updateEnemyBullet(): void
- updatePlaneBullet(): void
- updateBonuses(): void

Shooter Draw Item Manager

- shooterGameStatus: ShooterGameStatusFacade
- shooterPlaneBullets: List<ShooterPlaneBullet>
- shooterBonuses: List<ShooterBonus>
- healthAids: List<ShooterHealthAid>
- pointBuffs: LIst<ShooterPointBuff>
- shooterEnemyBullets: List<ShooterEnemyBullet>
- enemyExplosions: List<ShooterEnemyExplosion>
- planeExplosions: List<ShooterPlaneExplosion>
- shooterEnemies: List<ShooterEnemy>
- plane: ShooterPlane
- scorePaint: Paint
- levelPaint: Paint
- healthPaint: Paint
- canvas: Canvas

~ ShooterDrawItemManager(
shooterGameStatus ShooterGameStatusFacade)
~ setCanvas(canvas: Canvas): void
- setUpManager(): void
~ draw(): void
- setPaint(): void
- drawText(): void
- drawGameObject(): void
- drawExplosion(): void