

Identificador	Descripción de la acción de alto nivel			
Polis-001	Inicializa el juego con todos los elementos iniciales			
Pasos (usar pseudocódigo o similar)				
1.Inicializa el juego a partir de un fichero				
2.Inicializa los territorios				
3.Inicializa los mares				
4.Inicializa los puertos comerciales				
5.Inicializa los mercados				
6.Inicializa los proyectos				
7.Inicializa las polis				
8.Inicializa los eventos				
9.Inicializa la ronda				
10.Inicializa las tablas de comercio				
11.Inicializazción de los jugadores				
12.Inicializa el juego con todos los elementos				
Diagrama de estados (Opcional)				
Métodos de bajo nivel necesarios				
Paso	Clase	Método	Mem. Técn.	IU
1	PolReader	[] polReader()		NO
2	PolReader	[Map] readTerritoris()		NO
3	PolReader	[Map] readSeas()		NO
4	PolReader	[Map] readTradeDocks()		NO
5	PolReader	[Map] readMarkets()		NO
6	PolReader	[Map] readProjects()		NO
7	PolReader	[Map] readPolis()		NO
8	PolReader	[Map]readGameEvents()		NO
9	Round	[] Round()		NO
10	MarketChart	[MarkedChart]MarketChart()		NO
11	TextModeUI	[List]requestPlayersGame()		SI
12	Game	[]Game()		NO
Método de alto nivel				
[void] ElementsInitializer ()				
Diagrama de Colaboración (Opcional)				

Identificador	Descripción de la acción de alto nivel			
Polis-0002	Comprueba el final de la ronda			
Pasos (usar pseudocódigo o similar)				
1.Comprueba las polis asediadas 2.Comprueba los proyectos y quien recibe el prestigio 3.Comprueba alimentación de la población 4.Comprueba el crecimiento de la población 5.Comprueba Megalópolis 6.Reajuste de recursos perecederos 7.Comprueba Phoros 8.Inicializa siguiente ronda				
Diagrama de estados (Opcional)				
Métodos de bajo nivel necesarios				
Paso	Clase	Método	Mem. Técn.	IU
1	EndRoundManager	[void]checkSieges(Game: game,Player: player)		NO
2	EndRoundManager	[void]checkProjects(Player: player)		NO
3	EndRoundManager	[void]checkFeeding(Player :player)		NO
4	EndRoundManager	[void]checkGround(Player:player)		SI
5	EndRoundManager	[void]checkMegalopolis(Player :player)		NO
6	EndRoundManager	[void]checkGoodsAjust(Player: player)		NO
7	EndRoundManager	[void]checkPhoros(Player: player)		SI
8	EndRoundManager	[]initializeNextRound()	0003	NO
Método de alto nivel				
[void] EndRoundManager ()				
Diagrama de Colaboración (Opcional)				

Identificador	Descripción de la acción de alto nivel			
Polis-0003	Inicia la ronda del juego cargando y aplicando los proyectos.			
Pasos (usar pseudocódigo o similar)				
1. Se reparten 3 cartas de proyecto y se elige un evento.				
2. Se aplica evento.				
Diagrama de estados (Opcional)				
Métodos de bajo nivel necesarios				
Paso	Clase	Método	Mem. Técn.	IU
1	Round	[void]startRound()	0004	NO
2	Round	[void] startRound()		NO
Método de alto nivel				
void startRound()				
Diagrama de Colaboración (Opcional)				

Identificador	Descripción de la acción de alto nivel			
Polis-0004	Obtener una lista de elementos al azar.			
Pasos (usar pseudocódigo o similar)				
1. Se selecciona una lista de elementos al azar.				
Diagrama de estados (Opcional)				
Métodos de bajo nivel necesarios				
Paso	Clase	Método	Mem. Técn.	IU
1	RandomCollection	[List<T>]getRandomSubList (List<T> list, Integer elementCount)		NO
Método de alto nivel				
void RandomCollection()				
Diagrama de Colaboración (Opcional)				

Identificador	Descripción de la acción de alto nivel			
Polis-0005	Inicia la ronda del juego cargando y aplicando los proyectos.			
Pasos (usar pseudocódigo o similar)				
1. Se reparten 3 cartas de proyecto y se elige un evento.				
2. Se aplica evento.				
Diagrama de estados (Opcional)				
Métodos de bajo nivel necesarios				
Paso	Clase	Método	Mem. Técn.	IU
1	Round	[void]startRound()		NO
2	Round	[void] startRound()		NO
Método de alto nivel				
void startRound()				
Diagrama de Colaboración (Opcional)				

Identificador	Descripción de la acción de alto nivel			
Polis-0006	Finalizamos el juego resolviendo los parámetros necesarios para designar al ganador del juego.			
Pasos (usar pseudocódigo o similar)				
1. Comprueba si el jugador ha perdido la capital. 2. Comprueba el prestigio. 3. Comprobación del modo estándar de ganar. 4. Se declara el ganador del juego. 5. Finaliza juego.				
Diagrama de estados (Opcional)				
Métodos de bajo nivel necesarios				
Paso	Clase	Método	Mem. Téc.	IU
1	EndGameManager	[void] checkCapitals(Player:player)		NO
2	EndGameManager	[void] checkNoPrestige( Player:player)		NO
3	EndGameManager	[void] checkStandarEndGame( Player player1, Player player2)	0007	NO
4	EndGameManager	[Player] getWinner()		NO
5	EndGameManager	[void] endTheGame()		NO
Método de alto nivel				
void EndGameManager()				
Diagrama de Colaboración (Opcional)				

Identificador	Descripción de la acción de alto nivel			
Polis-0007	Comprobación del modo estándar para ganar , primero comprobando prestigio y luego recursos			
Pasos (usar pseudocódigo o similar)				
1. Obtención de prestigio 2. Obtencion de recursos				
Diagrama de estados (Opcional)				
Métodos de bajo nivel necesarios				
Paso	Clase	Método	Mem. Téc.	IU
1	EndGameManager	[int] getPlayerTotalPrestige(Player: player);		NO
2	EndGameManager	[int] getPlayerResourceCount(player )		NO
Método de alto nivel				
void EndGameManager()				
Diagrama de Colaboración (Opcional)				

Identificador	Descripción de la acción de alto nivel			
Polis-0008	Turnos, donde jugadores realizan sus dos acciones respectivamente.			
Pasos (usar pseudocódigo o similar)				
1. Jugador uno realiza sus acciones.				
2. Jugador dos realiza sus acciones.				
Diagrama de estados (Opcional)				
Métodos de bajo nivel necesarios				
Paso	Clase	Método	Mem. Técn.	IU
1	TextModeUI	[void]showAvailableActions(Game g, Player p)	0009	SI
2	TextModeUI	[void]showAvailableActions(Game g, Player p)	0009	SI
Método de alto nivel				
[void]TextModeUI ()				
Diagrama de Colaboración (Opcional)				

Identificador	Descripción de la acción de alto nivel			
Polis-0009	Turnos, donde jugadores realizan sus dos acciones respectivamente.			
Pasos (usar pseudocódigo o similar)				
1. Jugador hace una acción creación, la segunda acción no puede ser ésta. 2. Jugador hace una acción militar, la segunda acción no puede ser ésta. 3. Jugador hace una acción política, la segunda acción no puede ser ésta. 4. Jugador hace una acción pasa turno.				
Diagrama de estados (Opcional)				
Métodos de bajo nivel necesarios				
Paso	Clase	Método	Mem. Técn.	IU
1	creatorAction		0010	SI
2	MilitaryAction		0011	SI
3	politicAction		0012	SI
4	EmptyAction	[void] passTurn()		SI
Método de alto nivel				
[void]TextModeUI ()				
Diagrama de Colaboración (Opcional)				

Identificador		Descripción de la acción de alto nivel		
Polis-0010		Acciones de creación, Jugador decide cual realizar.		
Pasos (usar pseudocódigo o similar)				
1 Crear Hoplita 2 Crear Trirreme 3 Crear Próximo 4 Crear Barco de Comercio				
Diagrama de estados (Opcional)				
Métodos de bajo nivel necesarios				
Paso	Clase	Método	Mem. Técn.	IU
1	CreateAction	[boolean] createHoplite(Player owner, Polis polis, Round round)		SI
2	CreateAction	[boolean] createTrirreme(Player owner, Polis polis, Round round)		SI
3	CreateAction	[boolean] createProxenus(Player owner, Polis polis, Round round)		SI
4	CreateAction	[boolean] createTradeBoat(Player owner, Polis polis, Round round)		SI
Método de alto nivel				
[void]creatorAction()				
Diagrama de Colaboración (Opcional)				



Identificador		Descripción de la acción de alto nivel		
Polis-0011		Acciones militares, el Jugador decide cual realizar.		
Pasos (usar pseudocódigo o similar)				
1 Mover Hoplita. 2 Mover Trirreme. 3 Recaudar. 4 Asediar .				
Diagrama de estados (Opcional)				
Métodos de bajo nivel necesarios				
Paso	Clase	Método	Mem. Técn.	IU
1	MilitaryAction	[Boolean] moveHoplite(Player player, Round round,Territory initialPosition, Territory finalPosition, Integer numberOfUnits, Boolean multiMovement)		SI
2	MilitaryAction	[Boolean] moveTrirreme(Round round, Player player, Sea initialSea, Sea finalSea, Integer numberOfUnits, Boolean multiMovement)		SI
3	MilitaryAction	Boolean siegePolis(Player player,Position initialPosition, Polis siegedPolis )		SI
4	MilitaryAction	Boolean plunderTerritory(Player player)		SI
Método de alto nivel				
[void]MilitaryAction()				
Diagrama de Colaboración (Opcional)				

Identificador	Descripción de la acción de alto nivel			
Polis-0012	Acciones políticas, Jugador decide cual realizar			
Pasos (usar pseudocódigo o similar)				
1 Guerra Civil. 2 Mover Próximo. 3 Comenzar proyecto. 4 Comerciar.				
Diagrama de estados (Opcional)				
Métodos de bajo nivel necesarios				
Paso	Clase	Método	Mem. Téc.	IU
1	PoliticAction	[Boolean] civilWar(Player player, Polis designatedPolis)		SI
2	PoliticAction	[Boolean] moveProxenus(Player player, Polis destination)		SI
3	PoliticAction	[Boolean]startProject(Player player, Project project, Polis polis)		SI
4	PoliticAction	[Boolean] trade(Player player, Round round, MarketChart marketChart, Market market, String resource1, String resource2)		SI
Método de alto nivel				
[void]PoliticAction()				
Diagrama de Colaboración (Opcional)				

