File	Lines	Code	CVF	Severity	Category	Description	Recommendation	Client comment / Decision	Pull request / Commit	Status
DVP.sol	10–11	10import "@openzeppelin/contracts-upgradeable/token/ERC721/ERC721Upgradeable.sol"; 11import "@openzeppelin/contracts-upgradeable/token/ERC721/IERC721ReceiverUpgradeable.sol";	1	Minor	Procedural	These imports are not used.		The unused imports were removed	pull/18 / commit	Fixed
DVP.sol	34	34 address private potAddress;	2	Minor	Bad datatype	The type of this variable should be "IPOT".		Proposed changes were implemented	pull/18 / commit	Fixed
DVP.sol	44, 305	44 function initialize(address _potAddress) 305 function setPotAddress(address _potAddress)	3	Minor	Bad datatype	The type of the "_potAddress" argument should be "IPOT".		Proposed changes were implemented	pull/18 / commit	Fixed
DVP.sol	51–53	51 __Pausable_init(); 52 __Ownable_init(); 53 __UUPSUpgradeable_init();	4	Minor	Unclear behavior	Unchained functions should be used here.		Proposed changes were implemented.	pull/18 / commit	Fixed
DVP.sol	62	62 {}	5	Minor	Documentation	It is a good practice to put a comment into an empty block to explain why the block is empty.		Proposed changes were implemented by adding a comment inside the block	pull/18 / commit	Fixed
DVP.sol	67, 72, 77	67 event DeliveryConfirmed(uint256 indexed _tokenId, address _numAt); 72 event DeliveryExecuted(uint256 indexed _tokenId, address _numAt, address to); 77 event SettlementCanceled(uint256 indexed _tokenId, address _numAt, address to);	6	Minor	Suboptimal	All the parameters should be indexed.		Proposed changes were implemented	pull/18 commit	Fixed
DVP.sol	67, 72, 77	67 event DeliveryConfirmed(uint256 indexed _tokenId, address _numAt); 72 event DeliveryExecuted(uint256 indexed _tokenId, address _numAt, address to); 77 event SettlementCanceled(uint256 indexed _tokenId, address _numAt, address to);	7	Minor	Documentation	The semantics of the "_numAt" parameters is unclear.	Consider documenting.	The variable was renamed with a clearer name instead of adding documentation	pull/18 / commit	Fixed
DVP.sol	89–90, 111, 113, 116	89 address owner = IPOT(potAddress).ownerOf(tokenId); 90 IPOT.potStatus potStatus = IPOT(potAddress).getStatus(tokenId); 111 address assetTokenAddress = IPOT(potAddress).getDealDetailAddress(tokenId); 113 address receiver = IPOT(potAddress).getReceiver(tokenId); 116 uint256 numAssetTokensForSettlement = IPOT(potAddress).getDealDetailNum(tokenId);	8	Minor	Suboptimal	Several external calls to the same contract are performed in the same function. This could consume lots of gas.	Consider implementing a single getter function in the "IPOT" interface to fetch all the needed information in one call.	Declare a function getDetails (replace getDetailsForSettlement) and use it to fetch all the needed information in one call.	pull/18 commit	Fixed
DVP.sol	93	93 console.log("[DVP] DVP.checkDeliveryForPot(", tokenId, ")");	9	Minor	Suboptimal	This should be logged at the very beginning of the function..		Proposed changes were implemented	pull/18 commit	Fixed
DVP.sol	114, 117, 141, 185, 208, 252, 270	114 uint256 allowance = IERC20Upgradeable(assetTokenAddress).allowance(receiver, address(this)); 117 uint256 numAssetTokensOfReceiver = IERC20Upgradeable(assetTokenAddress).balanceOf(receiver); 141 IERC20Upgradeable(assetTokenAddress).transferFrom(receiver, address(this), numAssetTokensForSettlement); 185 uint256 numAssetTokensOfDvP = IERC20Upgradeable(assetTokenAddress).balanceOf(address(this)); 208 IERC20Upgradeable(assetTokenAddress).transfer(sender, numAssetTokensForSettlement); 252 uint256 numAssetTokensOfDvP = IERC20Upgradeable(assetTokenAddress).balanceOf(address(this)); 270 IERC20Upgradeable(assetTokenAddress).transfer(receiver, numAssetTokensForSettlement);	10	Minor	Bad datatype	The asset token doesn't have to be upgradeable.	Consider converting to "IERC20".	Proposed changes were implemented	pull/18 commit	Fixed
DVP.sol	114	114 uint256 allowance = IERC20Upgradeable(assetTokenAddress).allowance(receiver, address(this));	11	Minor	Suboptimal	Obtaining the allowance amount here seems redundant.	Just try to transfer tokens and the token contract will do proper allowance check.	The call to get the allowance and the check of this one was removed	pull/18 / commit	Fixed
DVP.sol	117	117 uint256 numAssetTokensOfReceiver = IERC20Upgradeable(assetTokenAddress).balanceOf(receiver);	12	Minor	Suboptimal	Obtaining the balance amount here seems redundant.	Just try to transfer tokens and the token contract will do proper balance check.	The call to get the balance and the check of this one was removed	pull/18 / commit	Fixed
DVP.sol	141, 208, 270	141 IERC20Upgradeable(assetTokenAddress).transferFrom(receiver, address(this), numAssetTokensForSettlement); 208 IERC20Upgradeable(assetTokenAddress).transfer(sender, numAssetTokensForSettlement); 270 IERC20Upgradeable(assetTokenAddress).transfer(receiver, numAssetTokensForSettlement);	13	Moderate	Flaw	The returned value is ignored.	Consider explicitly requiring the returned value to be true.	Proposed changes were implemented	pull/18 / commit	Fixed
DVP.sol	168, 177, 184, 187–188	168 IPOT.potStatus potStatus = IPOT(potAddress).getStatus(tokenId); 177 address owner = IPOT(potAddress).ownerOf(tokenId); 184 address assetTokenAddress = IPOT(potAddress).getDealDetailAddress(tokenId); 187 uint256 numAssetTokensForSettlement = IPOT(potAddress).getDealDetailNum(tokenId); 188 address sender = IPOT(potAddress).getSender(tokenId);	14	Minor	Suboptimal	Several external calls to the same contract are performed in the same function. This could consume lots of gas.	Consider implementing a single getter function in the "IPOT" interface to fetch all the needed information in one call.	Declare a function getDetailsForDelivery and use it to fetch all the needed information in one call.	pull/18 commit	Fixed
DVP.sol	185	185 uint256 numAssetTokensOfDvP = IERC20Upgradeable(assetTokenAddress).balanceOf(address(this));	15	Minor	Suboptimal	Obtaining the balance amount here seems redundant.	Just try to transfer tokens and the token contract will do proper balance check.	The call to get the balance and the check of this one was removed	pull/18 commit	Fixed
DVP.sol	242, 248, 251, 254, 269	242 IPOT.potStatus potStatus = IPOT(potAddress).getStatus(tokenId); 248 address owner = IPOT(potAddress).ownerOf(tokenId); 251 address assetTokenAddress = IPOT(potAddress).getDealDetailAddress(tokenId); 254 uint256 numAssetTokensForSettlement = IPOT(potAddress).getDealDetailNum(tokenId); 269 address receiver = IPOT(potAddress).getReceiver(tokenId);	16	Minor	Suboptimal	Several external calls to the same contract are performed in the same function. This could consume lots of gas.	Consider implementing a single getter function in the "IPOT" interface to fetch all the needed information in one call.	Declare a function getDetailsForSettlement and use it to fetch all the needed information in one call.	pull/18 commit	Fixed
DVP.sol	252	252 uint256 numAssetTokensOfDvP = IERC20Upgradeable(assetTokenAddress).balanceOf(address(this));	17	Minor	Suboptimal	Obtaining the balance amount here seems redundant.	Just try to transfer tokens and the token contract will do proper balance check.	The call to get the balance and the check of this one was removed	pull/18 commit	Fixed
DVP.sol	287, 293	287 IPOT.potStatus potStatus = IPOT(potAddress).getStatus(tokenId); 293 uint256 mintTime = IPOT(potAddress).getMintTime(tokenId);	18	Minor	Suboptimal	Several external calls to the same contract are performed in the same function. This could consume lots of gas.	Consider implementing a single getter function in the "IPOT" interface to fetch all the needed information in one call.	Declare a function getStatusAndMintTime and use it to fetch all the needed information in one call.	pull/18 commit	Fixed
DVP.sol	305	305 function setPotAddress(address _potAddress)	19	Minor	Suboptimal	This function should emit some event.		Proposed changes were implemented Emit the event POTAddressChanged	pull/18 commit	Fixed
DVP.sol	318	318 returns (address)	20	Minor	Bad datatype	The return type should be "IPOT".		The return type was modified for IPOT	pull/18 commit	Fixed
IPOT.sol	2	2pragma solidity >=0.8.12 <0.9.0;	21	Minor	Procedural	Consider specifying as "^0.8.12" or "^0.8.0".		The import is fixed to the version 0.8.17 to be consistent with DVP.sol	pull/18 commit	Fixed
IPOT.sol	4	4import "@openzeppelin/contracts/token/ERC721/extensions/ERC721Pausable.sol";	22	Minor	Procedural	Should be: import "@openzeppelin/contracts/token/ERC721/IERC721.sol";		Proposed changes were implemented	pull/18 commit	Fixed
IPOT.sol	15, 18, 21, 26, 31, 36, 41, 46	15 function initiatePayment(uint256 tokenId) 18 function deactivatePot(uint256 tokenId) 21 function getMintTime(uint256 _tokenId) 26 function getStatus(uint256 _tokenId) 31 function getSender(uint256 _tokenId) 36 function getReceiver(uint256 _tokenId) 41 function getDealDetailNum(uint256 _tokenId) 46 function getDealDetailAddress(uint256 _tokenId)	23	Minor	Readability	In some cases argument names are prefixed with underscore, while in other cases they are not.	Consider using a consistent naming schema.	Add a prefix with underscore to all arguments	pull/18 commit	Fixed
IPOT.sol	15, 18	15 function initiatePayment(uint256 tokenId) 18 function deactivatePot(uint256 tokenId)	24	Minor	Procedural	These functions should emit some events and these events should be declared in this interface.		Move the events from POT.sol into IPOT.sol	pull/18 commit commit 2	Fixed