LandPlot + Marketplace

Inspect the project repository(project.zip) and then implement following improvements.

LandPlot(NFT) contract improvements:

1. Ensure that NFT with ID 0 and Coordinate 0,0 is minted to the contract owner.
2. Getting the coordinates for an NFT that has not yet been minted should be 0,0
3. Getting the NFT for which there are no matching coordinates should fail
4. minting function should be rewritten to ensure that duplicates cannot be written
5. rename the stupid variables (for some reason \_worldSize is the amount of minted nfts, i was crazy or something)
6. Change the admin minting function and the user minting function such that they use the same underlying implementation, simply a different user
7. Add a “multitransfer” function that allows the user to send more than 1 Plot to an address at once.
8. Create ILandPlot.sol, and in general reformat/rewrite contract to fit common solidity standards
9. Test should ensure that all rules for token minting is being followed (priced being paid, size within constraints)

Simple LandPlot Auction House Contract

1. User looking to sell should be able to put any ERC721 compliant token up for auction.
2. User looking to sell should be able to put multiple ERC721 compliant tokens up for auction as a ‘bundle’ for a single price
3. User looking to sell can cancel their auction at any time, returning all eth to bidders.
4. User looking to buy should be able to see the ids of tokens that are up for auction
5. User looking to buy should be able to see an array of ids of a token bundle that is up for auction
6. User looking to buy should be able to see information about the auction (current price, blocks remaining, who the current highest bidder is)
7. User looking to buy should be able to bid using with a payable function
8. The ETH of the highest bid is locked up, but if another user outbids the previous highest bidder, the eth is automatically returned.
9. Once the auction ends (period specified initially by the user selling), the eth is automatically sent to the sellers wallet and NFT to the winning bidder.
10. Additionally, a user may put out a “bid” to purchase a specific ERC721 by inputting the ERC721 contract address, tokenId, and expiration time (max 1 week), along with sending some eth. The owner of that token may then, before the week expires, claim that “bid” and receive the eth, sending the ERC721 token off to the bidder.
11. Create IMarketplace.sol, or whatever you call it all normal standards ofc
12. Test should ensure that funds and tokens are all sent as they should in multiple scenarios (auction closed, auction cancel, people bidding in the same block, etc)

Both contracts should be Upgradable & behind proxies owned by the deployer

Both contracts should also have a deployment & upgrade test scripts written, along with tests that test the basic functionality of every method at least once