I will preface this by saying that I rushed some areas of the test since it wasn't mentioned in the requirements.For example a proper Input handling solution or event propagation system. And as such some components are coupled. Also I've done this type of inventory a few times so it is based on past projects.

The inventory works as suggested. It is based on scriptable objects (ItemDefinition). It is stored in a struct that handles the quantity. The Inventory class handles everything related to the inventory.
UIInventory handles the refreshing of the inventory slots.
Both ui and data inventories have slots.For the ui part I used unitys interfaces for dragging, dropping, splitting and using items. For the data part it just stores the item and index and some simple operations.
For saving I simply convert some of the data structures to a serializable counterpart. The scriptable object is stored by name and on load it searches for that name.

The movement is as basic as it gets. Same for the camera, I used Cinemachine. The character/animation is from the asset store.
The Collectable class handles the collecting. The implementation is not even close to ideal but for this test it seemed like enough.

I spent most of the time making sure the Inventory system works but considering the time invested in this I think it is ok. Unfortunately I couldn't spare time for polish. I should have taken the task in the weekend.