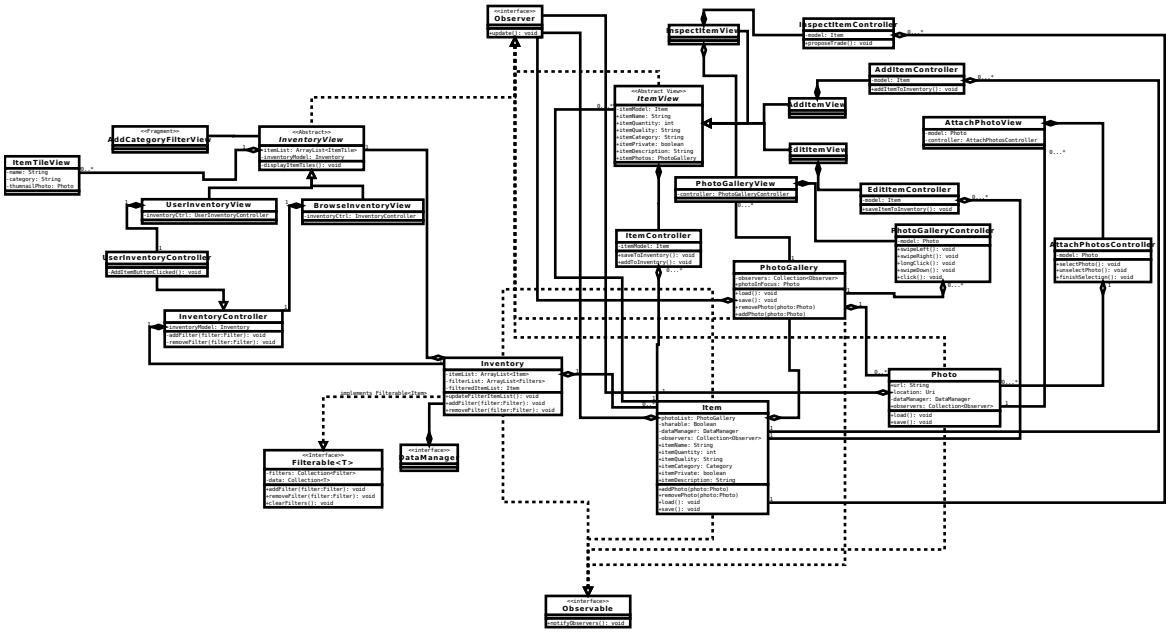
Observer
-update(): void
InspectItemView
InspectItemController
-model: Item
-proposeTrade(): void
AddItemController
-model: Item
-addItemToInventory(): void
<<Abstract View>>
ItemView
-itemModel: Item
-itemName: String
-itemQuantity: int
-itemQuality: String
-itemCategory: String
-itemPrivate: boolean
-itemDescription: String
-itemPhotos: PhotoGallery
AddItemView
EditItemView
AttachPhotoView
-model: Photo
-controller: AttachPhotosController
<<fragment>>
AddCategoryFilterView
<<Abstract>>
InventoryView
-itemList: ArrayList<ItemTile>
-inventoryModel: Inventory
-displayItemTiles(): void
ItemTileView
-name: String
-category: String
-thumnailPhoto: Photo
UserInventoryView
-inventoryCtrl: UserInventoryController
BrowseInventoryView
-inventoryCtrl: InventoryController
PhotoGalleryView
-controller: PhotoGalleryController
EditItemController
-model: Item
-saveItemToInventory(): void
PhotoGalleryController
-model: Photo
-swipeLeft(): void
-swipeRight(): void
-longClick(): void
-swipeDown(): void
-click(): void
AttachPhotosController
-model: Photo
-selectPhoto(): void
-unselectPhoto(): void
-finishSelecting(): void
ItemController
-itemModel: Item
-saveToInventory(): void
-addToInventory(): void
UserInventoryController
-addItemButtonClicked(): void
PhotoGallery
-observers: Collection<Observer>
-photoInFocus: Photo
-load(): void
-save(): void
-removePhoto(photo:Photo)
-addPhoto(photo:Photo)
InventoryController
-inventoryModel: Inventory
-addFilter(filter:Filter): void
-removeFilter(filter:Filter): void
Inventory
-itemList: ArrayList<Item>
-filterList: ArrayList<Filters>
-filteredItemList: Item
-updateFilteredItemList(): void
-addFilter(filter:Filter): void
-removeFilter(filter:Filter): void
Photo
-uri: String
-location: Uri
-dataManager: DataManager
-observers: Collection<Observer>
-load(): void
-save(): void
Item
-photoList: PhotoGallery
-sharable: Boolean
-dataManager: DataManager
-observers: Collection<Observer>
-itemName: String
-itemQuantity: int
-itemQuality: String
-itemCategory: Category
-itemPrivate: boolean
-itemDescription: String
-addPhoto(photo:Photo)
-removePhoto(photo:Photo)
-load(): void
-save(): void
<<Interface>>
Filterable<T>
-filters: Collection<Filter>
-data: Collection<T>
-addFilter(filter:Filter): void
-removeFilter(filter:Filter): void
-clearFilters(): void
<<Interface>>
DataManager
<<Interface>>
Observable
-notifyObservers(): void