



Complete Implementation Checklist

Files to Update (In Order)

✓ 1. Item.java

ADD these two methods:

```
java
```

```

/**
 * ★ NEW: Determine which gear slot this item can be equipped in
 */
public String getEquipSlot() {
    if (isWeapon()) return "WEAPON";

    if (isArmor()) {
        String nameLower = name.toLowerCase();
        if (nameLower.contains("helmet") || nameLower.contains("hat")) return "HEAD";
        if (nameLower.contains("pants") || nameLower.contains("leggings")) return "PANTS";
        if (nameLower.contains("gloves") || nameLower.contains("gauntlet")) return "GLOVES";
        if (nameLower.contains("boots") || nameLower.contains("shoes")) return "SHOES";
        return "TOP_ARMOR"; // Default armor goes to chest
    }

    if (isAccessory()) {
        String nameLower = name.toLowerCase();
        if (nameLower.contains("earring")) return "EARRINGS";
        if (nameLower.contains("necklace") || nameLower.contains("amulet")) return "NECKLACE";
        if (nameLower.contains("bracelet")) return "BRACELET";
        if (nameLower.contains("ring")) return "RING_1";
        if (nameLower.contains("boots") || nameLower.contains("shoes")) return "SHOES";
        return "SPECIAL";
    }

    return null;
}

/**
 * ★ NEW: Can this item be equipped?
 */
public boolean isEquippable() {
    return getEquipSlot() != null;
}





```

Where: Add at the bottom of Item.java, before the closing brace

✓ 2. UIInventorySlot.java

ALREADY DONE if using the debug version!

Contains:

-  NEW badge system
-  `markAsNew()` method
-  `update()` method for animation
-  `drawNewItemBadge()` rendering




UPDATE the `onRightClick()` method:

Replace your existing `onRightClick()` with the version from artifact "UIInventorySlot - Right-Click to Equip (All Gear)"

3. UIGearSlot.java

REPLACE with the icon-enabled version from artifact "UIGearSlot.java (Updated with Icons)"

Contains:

-  Icon rendering (same as inventory)
 -  Rarity borders
 -  Fallback placeholders
-

4. UIScrollableInventoryPanel.java

UPDATE `addItemToCurrentTab()` method:

Replace the existing method with BOTH methods from artifact "UIScrollableInventoryPanel - Add NEW Badge Support":

```
java

// Method 1: With markAsNew parameter
public boolean addItemToCurrentTab(Item item, boolean markAsNew) { ... }

// Method 2: Backwards compatible (calls method 1)
public boolean addItemToCurrentTab(Item item) {
    return addItemToCurrentTab(item, true);
}
```

Where: Find your existing `addItemToCurrentTab()` method and replace it with both versions

✓ 5. UIManager.java

UPDATE addItemToInventory() method:

Replace with:

```
java

/**
 * ★ UPDATED: Add item to inventory (marks as new by default)
 */
public boolean addItemToInventory(Item item) {
    return addItemToInventory(item, true);
}

/**
 * ★ NEW: Add item with optional "NEW" badge
 */
public boolean addItemToInventory(Item item, boolean markAsNew) {
    if (inventoryContainer == null) {
        createInventorySystem();
    }

    if (inventoryGrid != null) {
        boolean added = inventoryGrid.addItemToCurrentTab(item, markAsNew);

        if (added) {
            System.out.println("Added item to inventory (tab=" + currentInventoryTab +
                               ", marked as new=" + markAsNew + ")");
            notifyInventoryUpdate();
            return true;
        }
    }

    System.out.println("Inventory full!");
    return false;
}
```

Testing Checklist

Test 1: Icons Show Up

```
java

// Add test items
uiManager.addItemToInventory(ItemManager.createWoodenShortSword());
uiManager.addItemToInventory(ItemManager.createLeatherArmor());
uiManager.addItemToInventory(ItemManager.createPowerRing());
```

Expected:

- ☒ Icons appear in inventory (or colored placeholders with letters)
 - ☒ Rarity-colored borders around items
 - ☒ Green "NEW" badges on all three items
-

Test 2: Right-Click Equipping

```
java

// Open inventory (I key)
// Right-click the Wooden Short Sword
```

Expected:

- ☒ Sword moves to Weapon slot
 - ☒ Icon appears in Weapon slot
 - ☒ Sword removed from inventory
 - ☒ Console: "☒ Equipped Wooden Short Sword to WEAPON slot"
-

Test 3: All Gear Types

```
java

// Right-click armor
// Right-click ring
// Right-click boots
```

Expected:

- ☒ Armor → Top Armor slot
 - ☒ Ring → Ring 1 slot (or Ring 2 if Ring 1 occupied)
 - ☒ Boots → Shoes slot
 - ☒ Icons show in all gear slots
-

Test 4: NEW Badge Animation

```
java

// Add an item
uiManager.addItemToInventory(ItemManager.createHealthPotion());
// Watch for 5 seconds
```

Expected:

- ☒ Green "NEW" badge appears immediately
 - ☒ Badge pulses/glows
 - ☒ Badge fades away after 5 seconds
-

Test 5: Stackable Items

```
java

// Add same item multiple times
for (int i = 0; i < 5; i++) {
    uiManager.addItemToInventory(ItemManager.createCarvedWood());
}
```

Expected:

- ☒ Items stack into one slot
 - ☒ Shows count "5" in bottom-right
 - ☒ NEW badge shows on stacked item
-


Troubleshooting

Compilation Errors

Error: "Cannot find symbol: getEquipSlot()"

-  **Fix:** Add the two methods to Item.java

Error: "Cannot find symbol: markAsNew()"




-  **Fix:** Use updated UIInventorySlot.java (debug version)

Error: "Method does not override or implement a method from a supertype"




-  **Fix:** Check that onRightClick() is in UIInventorySlot, not UIScrollableInventoryPanel
-

Runtime Errors




Icons don't show:

-  Check console for icon loading messages
-  Verify PNG files are in `resources/items/icons/`
-  Rebuild project



NEW badge doesn't appear:


-  Check `markAsNew` parameter is true
-  Verify `update()` is being called on slots
-  Check console for "🔥 Marked slot X as NEW"

Can't equip items:

-  Verify Item.java has `getEquipSlot()` method
-  Check item names contain keywords ("sword", "armor", "ring", etc.)
-  Look at console for error messages









Wrong gear goes to wrong slot:

-  Check item names in ItemManager
-  Verify logic in `getEquipSlot()` method

-  Ensure item types are correct (WEAPON, ARMOR, ACCESSORY)
-

Success Criteria

Your implementation is complete when:

1.  **Icons show** in inventory slots
 2.  **Icons show** in gear slots when equipped
 3.  **Colored borders** match item rarity
 4.  **Right-click** any gear to auto-equip
 5.  **Green "NEW" badge** shows on new items
 6.  **Badge fades** after 5 seconds
 7.  **Stack counts** show on consumables
 8.  **Tooltips** show item stats on hover
-

Quick Reference

Add item with badge:

```
java  
  
addItemToInventory(item); // Shows NEW badge
```

Add item silently:

```
java  
  
addItemToInventory(item, false); // No badge
```

Manual badge control:

```
java  
  
slot.markAsNew(); // Show badge  
slot.clearNewBadge(); // Remove badge
```

Final Notes

- All updates maintain backwards compatibility
- Existing code continues to work
- NEW badges are optional (controlled by boolean parameter)
- System works with or without icon files (fallback placeholders)

You're all set! 