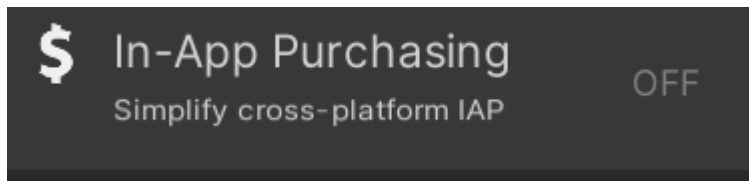


### **Practical 13 – In-App Purchase**

1. Create a new **Unity Project**.
2. From the **Windows > Services**, turn on In-App Purchasing by selecting the **“OFF”**.



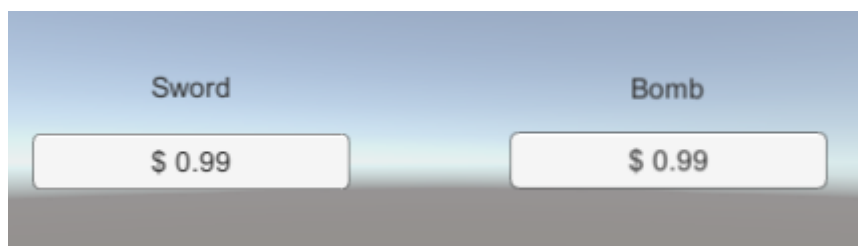
3. Once selected, a project settings window will be displayed.
  - a) Select organization (your account name), follow by **“Create Project ID”**.
  - b) Next a question “Will this app be primarily targeted to children under age 13”, select **“No”**. Select Save.

<https://docs.unity3d.com/Manual/UnityAnalyticsCOPPA.html>

4. Select **“OFF”** from the In-App Purchasing Settings to download the SDK.
5. Close **Project Settings** Window.

### **Setup IAP Button**

1. From the Hierarchy Panel, create a **UI>Panel**.
2. Create two IAP Buttons. To do so, go to **Windows> Unity IAP > Create IAP Button**.
  - a) Create button to purchase **“sword”**. To do so, select IAP Button from the Hierarchy Panel, change to **“sword”**.
  - b) Select the Text, change to **\$0.99**.
  - c) Create a label for the sword. To do so, add a new **UI> Text**. Rename the Text object as **“sword text”**. From the Inspector Panel, change text to **“Sword”**.
  - d) Repeat step a) to d) to create the IAP Button for **“Bomb”**

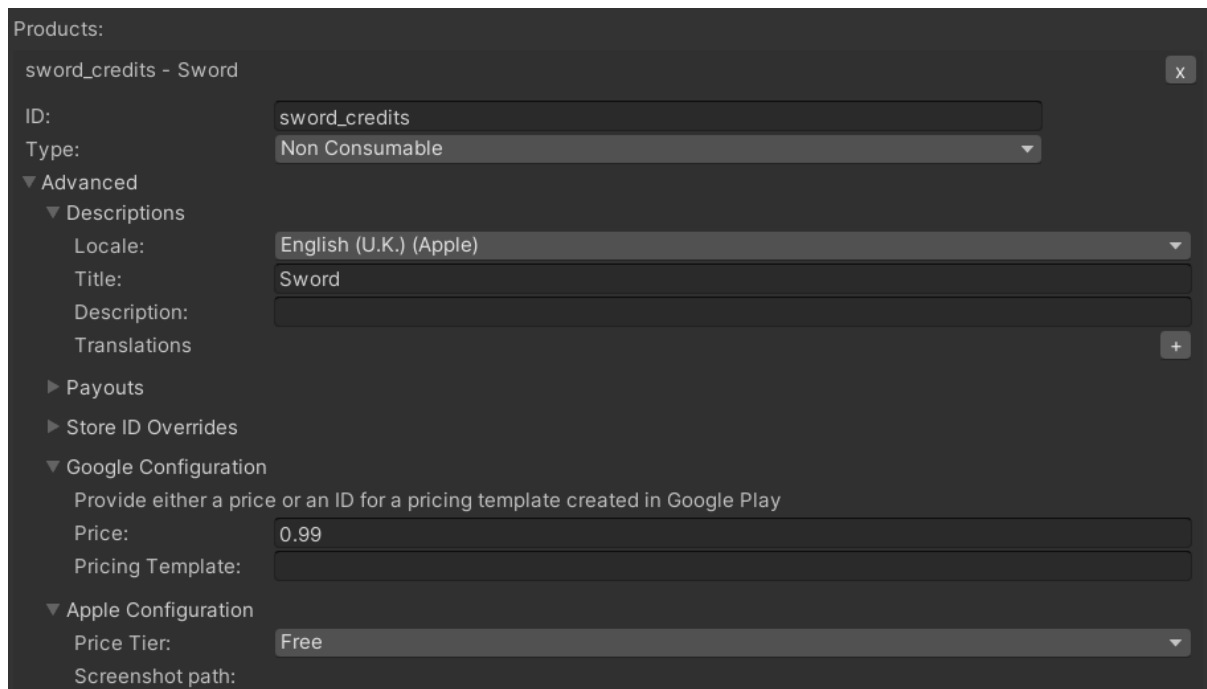


## Setup IAP Catalog

1. Select IAP Button “**sword**” from the Hierarchy panel.
2. From the Inspector Panel, select **IAP Catalogue**.

<https://docs.unity3d.com/Manual/UnityIAPDefiningProducts.html>

3. Add **sword** and **bomb** products into the **catalogue**. Follow settings below to create **catalogue**.



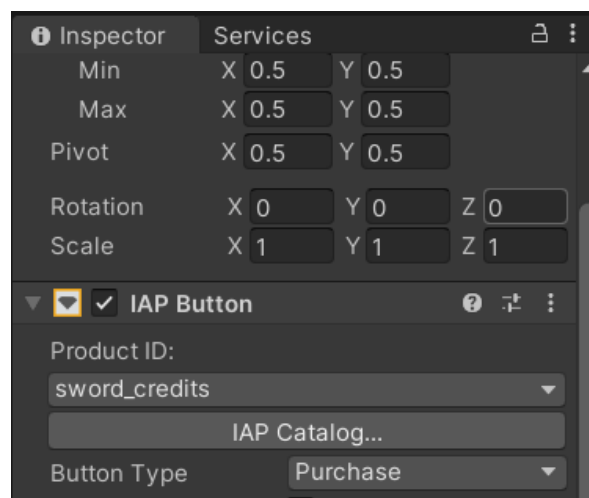
The screenshot shows the Unity Inspector panel for a product named "sword\_credits - Sword". The settings are as follows:

- ID:** sword\_credits
- Type:** Non Consumable
- Advanced**
  - Descriptions**
    - Locale:** English (U.K.) (Apple)
    - Title:** Sword
    - Description:** (empty)
    - Translations:** (+)
  - Payouts**
  - Store ID Overrides**
  - Google Configuration**
    - Provide either a price or an ID for a pricing template created in Google Play
    - Price:** 0.99
    - Pricing Template:** (empty)
  - Apple Configuration**
    - Price Tier:** Free
    - Screenshot path:** (empty)

4. Select “**Add Product**” to add “bomb”.
5. Check the selection “Automatically initialize **UnityPurchasing**” selection.

Automatically initialize UnityPurchasing (recommended) ☒

6. From the Inspector Panel, add each **ID** to the button respectively.



The screenshot shows the Unity Inspector panel for an IAP Button. The settings are as follows:

- Inspector**
  - Min:** X 0.5, Y 0.5
  - Max:** X 0.5, Y 0.5
  - Pivot:** X 0.5, Y 0.5
  - Rotation:** X 0, Y 0, Z 0
  - Scale:** X 1, Y 1, Z 1
- IAP Button**
  - Product ID:** sword\_credits
  - IAP Catalog...** (empty)
  - Button Type:** Purchase

### Scripting for Weapon Purchase

1. From the Panel, create a new script “**weapon purchase**”.
2. Import library

```
UnityEngine.Purchasing;
```

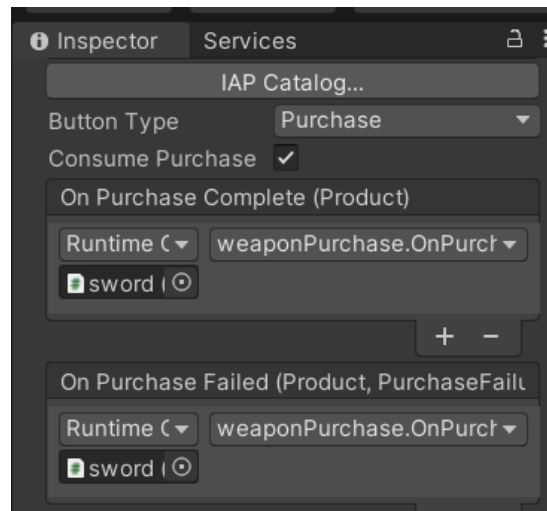
3. Declare variable as following

```
public GameObject iapButton;  
GameController gc;  
float weaponValue = 0.99f;
```

4. Create methods OnPurchaseComplete and OnPurchaseFailure. Add the following codes to the script.

```
public void OnPurchaseComplete(Product product)  
{  
#if UNITY_EDITOR  
    StartCoroutine(UpdateCredits());  
#else  
#endif  
}  
  
public void OnPurchaseFailure(Product product, PurchaseFailureReason reason)  
{  
    Debug.Log("Purchase of product" + product.definition.id + "fail due to:"  
+ reason);  
}  
  
private IEnumerator UpdateCredits()  
{  
    yield return new WaitForEndOfFrame();  
    gc =  
    GameObject.FindGameObjectWithTag("GameController").GetComponent<GameController>()  
    ;  
    gc.updateCredits(weaponValue); }  
}
```

5. From the Inspector Panel, drag and drop Weapon Purchase Script to the “sword” and “bomb” IAP button.



6. Create a GameController empty object and GameController script with the tag "GameController". Add the method updateCredits update the price.

6. Hit Play. You should be able to perform purchase weapon action now.