

# bada Tutorial: Commerce



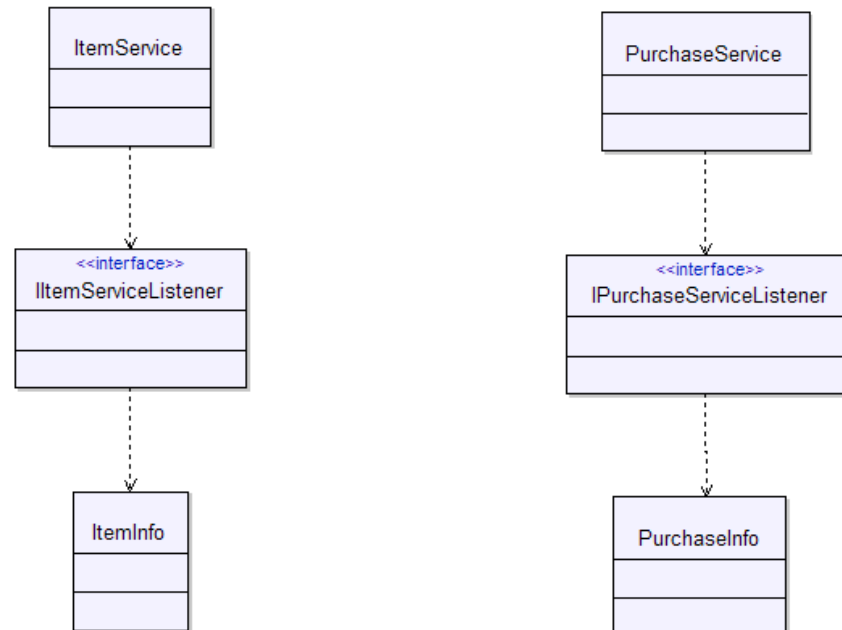
# Contents

- Essential Classes and Relationships
- Overview
- Sales Process Flow
- Development Process Flow
- Entering Items
- ItemService
- ItemInfo
  - Example: Get a List of Items
- PurchaseService
- PurchaseInfo
  - Example: Make a Sale
  - Example: Get Purchase Information
- Sample Application: “InAppBuying”
- FAQ
- Review
- Answers



# Essential Classes and Relationships

Feature	Provided by
Provides item management operations.	ItemService
Provides purchase operations.	PurchaseService
Contains purchase Information.	PurchaseInfo
Contains item Information.	ItemInfo
Provides a listener for ItemService operations.	IItemServiceListener
Provides a listener for PurchaseService operations.	IPurchaseServiceListener



# Overview

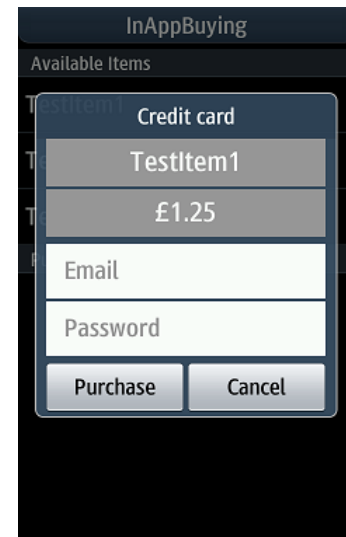
- The Commerce::Store namespace lets you sell items through your application at Samsung Apps after you register them in Samsung Apps Seller Office.
- Key features include:
  - Getting Item information from Samsung Apps.
  - Purchasing an item in your application.
  - Retrieving list of purchased items.



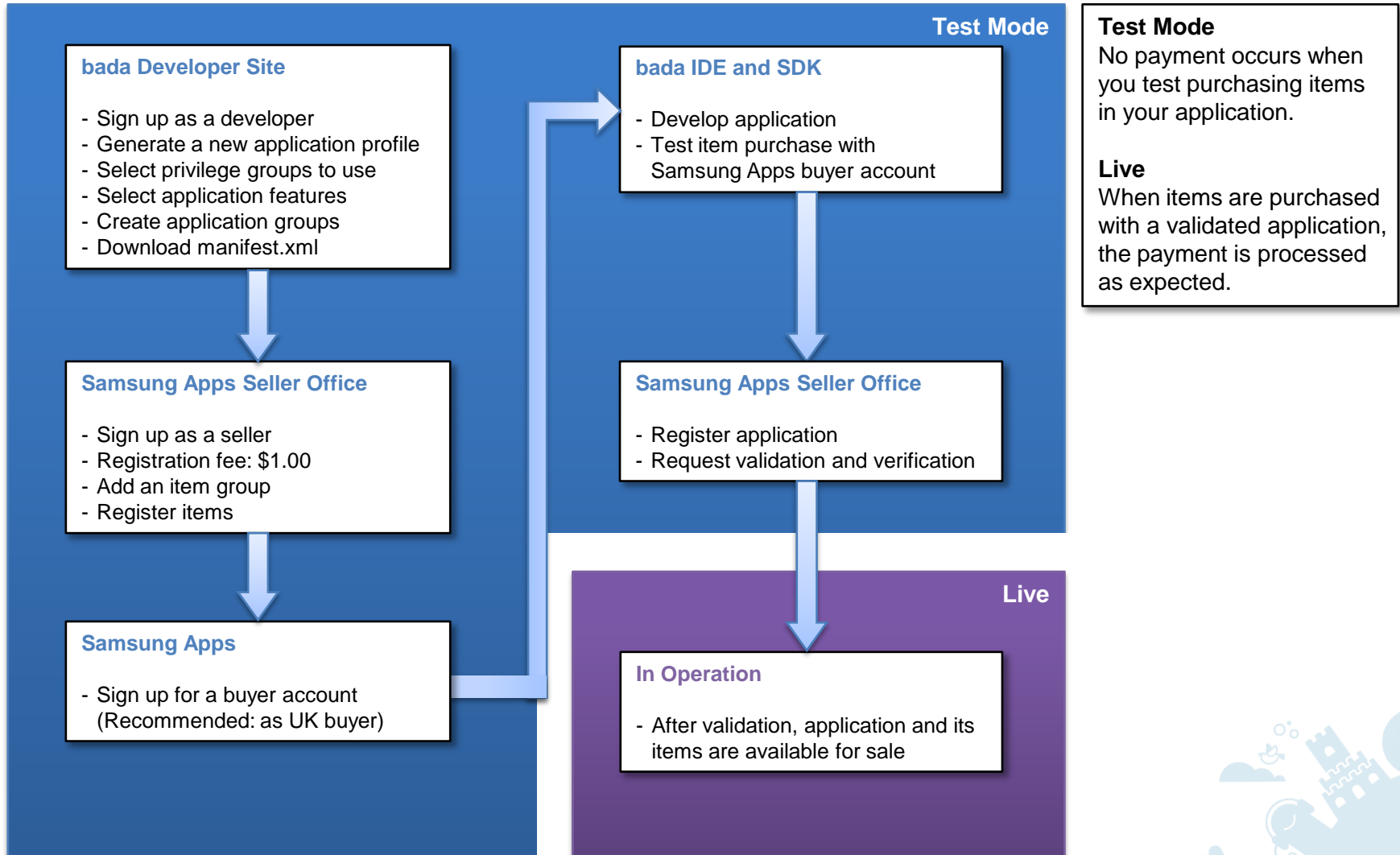
# Sales Process Flow

There are 4 basic steps to the sales process:

1. Enter your items into Samsung Apps Seller Office. This may require testing, verification, or authentication before your item is accepted and goes live.
2. In your application, get the list of items available through Samsung Apps and display them for your customer to see.
3. Let your customer choose an item and request payment at Samsung Apps.  
The available payment methods vary depending on the country. However, the credit card payment is supported for most of the countries. For more information, see the [bada developer site](#).
4. Retrieve your customer's purchase.



# Development Process Flow



# Entering Items (1/2)

- Go to Samsung Apps Seller Office (<http://seller.samsungapps.com/>) and, in the menu, select **Applications > Management > Item** to submit your items for possible inclusion.
- First create an item group and add items in it.  
Note that you must remember the item group ID to use it as a parameter for the APIs later.
- Before registering the application in Samsung Apps Seller Office (**Applications > Add New Application**), you can test purchasing items in your application without actual payment (test mode).
- When you add your application, you can register your items to the application by selecting the item group ID.



# Entering Items (2/2)

You can enter items in Samsung Apps Seller Office (<http://seller.samsungapps.com/>).

The screenshot shows the Samsung Apps Seller Office interface. At the top, there is a navigation bar with tabs for 'My Profile', 'Buyer Support', 'Applications' (selected), 'Accounting', 'Statistics', and 'Support'. The main content area is titled 'Item' and shows a list of 3 registered item groups. A table lists the item groups with columns for Item Group Title, Item Group ID, Number of items, Application Title, Application Status, and Edit Item Group. The table contains three rows: 'IG InAppBuying' (000268436235, 3ea, InAppBuying, Registration Status), 'IG Image Bada' (000268436321, 1ea, Image Bada, Registration Status), and 'New InAppBuying' (100000000006, 3ea, empty, empty). There are 'Edit' buttons for each row. A 'Delete' button is also visible. The left sidebar contains 'APPLICATIONS' (Sales Status, Approval Status, Registration Status, Discounts Status), 'Add New Application', 'MANAGEMENT' (UID, Widget ID, Item), and 'INSTRUCTIONS FOR USE' (User Guide, Guide to Request Commercial Seller Status, Guide to Uploading Large Files, In App Purchase Guideline).

**APPLICATIONS**

- Sales Status
- Approval Status
- Registration Status
- Discounts Status

**MANAGEMENT**

- UID
- Widget ID
- Item**

**INSTRUCTIONS FOR USE**

- User Guide
- Guide to Request Commercial Seller Status
- Guide to Uploading Large Files
- In App Purchase Guideline

**Item** HOME > Applications > Item

3 Item Group(s) are registered.

If you want to charge for your product and Item, you can request commercial seller Status. [Go](#) [Add Item Group](#)

	Item Group Title	Item Group ID	Number of items	Application Title	Application Status	Edit Item Group
<input type="checkbox"/>	<a href="#">IG InAppBuying</a>	000268436235	3ea	InAppBuying	Registration Status	<a href="#">Edit</a>
<input type="checkbox"/>	<a href="#">IG Image Bada</a>	000268436321	1ea	Image Bada	Registration Status	<a href="#">Edit</a>
<input type="checkbox"/>	<a href="#">New InAppBuying</a>	100000000006	3ea			<a href="#">Edit</a>

[Delete](#) 1

# ItemService

- `ItemService` lets you retrieve a list of items that you have previously registered in Samsung Apps Seller Office.
- Create an `ItemService` instance, and call `GetItemList()` using the item group ID that you received when you registered your items in Samsung Apps Seller Office.
- Get the item list in the `IItemServiceListener` as an `IList` of `ItemInfo` objects.



# ItemInfo

- Your items are returned as an `IList` of `ItemInfo` objects. Iterate over the list to display them to your customers.
- `ItemInfo` contains information about your item as shown below.

Property	Description
ID	Item's unique ID in Samsung Apps.
Name	Item name or title.
Description	Brief description of the item.
Price	Price of the item.
Currency	Store currency, such as: \$, ¢, £, ¥, ₩, €
Image URL	Image URL for the item.
Download URL	URL to download the item.
Reserved 1	Reserved.
Reserved 2	Reserved.

# Example: Get a List of Items

Get a list of items to display to your potential customers.

- Open `\<BADA_SDK_HOME>\Examples\Commerce\src\ItemServiceExample.cpp, GetItemListExample()`

1. Implement an `IItemServiceListener`.

2. Create an `IItemServiceListener` instance.

3. Create an `ItemService` instance.

4. Construct the `ItemService` with the listener:

```
ItemService::Construct(listener)
```

5. Get the list from the server:

```
ItemService::GetItemList(itemGroupId, start, end, reqId)
```

```
IItemServiceListener::OnItemListReceived()
```

6. Iterate over the results and read each `ItemInfo` object.



# PurchaseService

- Once your customer has decided what they want to purchase, send them to Samsung Apps to complete their purchase.
  - `PurchaseService::PurchaseItem(appId, itemId, reqId)`
- Check that the purchase was completed in the `IPurchaseServiceListener`'s `OnItemPurchaseCompleted()` method.
- To check the records of purchases were made on the device, call the `PurchaseService`'s `GetPurchaseInfo()` method.
  - `GetPurchaseInfo(itemGroupId, paymentId†, startDate†, endDate†, startNum, endNum)`

<sup>†</sup> Optional. Pass null to ignore these parameters.



# PurchaseInfo

Purchase information is returned to you as an `IList` of `PurchaseInfo` objects. Iterate over the list to find the specific purchase record that you are looking for.

Property	Description
Item ID	Item's unique ID in Samsung Apps.
Item name	Item name or title.
Purchase date	Transaction date.
Price	Price paid.
Currency	Transaction currency.
Item description	Brief description of the item.
Image URL	Image URL for the item.
Download URL	URL to download the item.
Payment ID	Transaction ID.
Reserved 1	Reserved.
Reserved 2	Reserved.

# Example: Make a Sale

Make a sale through the application.

- Open `\<BADA_SDK_HOME>\Examples\Commerce\src`  
`\PurchaseServiceExample.cpp, PurchaseItemExample()`

1. Implement an `IPurchaseServiceListener`.
2. Create an instance of `IPurchaseServiceListener`.
3. Create an instance of `PurchaseService`.
4. Construct the `PurchaseService` with the listener:  
`PurchaseService::Construct(listener)`
5. Send the customer to Samsung Apps to complete the purchase:  
`PurchaseService::PurchaseItem(itemGroupId, itemId, reqId)`  
`IPurchaseServiceListener::OnItemPurchaseCompleted()`
6. Read the purchase results from the listener method.



# Example: Get Purchase Information

Get information about purchases that were made on the device.

– Open `\<BADA_SDK_HOME>\Examples\Commerce\src  
\PurchaseServiceExample.cpp, GetPurchaseInfoExample()`

1. Implement an `IPurchaseServiceListener`.

2. Create an instance of the listener.

3. Create a `PurchaseService` instance.

4. Construct the `PurchaseService` with the listener:

```
PurchaseService::Construct(listener)
```

5. Get the purchase information from the server:

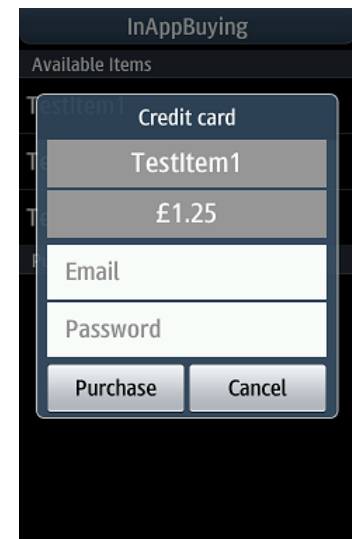
```
PurchaseService::GetPurchaseInfo(itemGroupId,  
paymentId, null, null, 1, 2)
```

```
IPurchaseServiceListener::OnPurchaseInfoReceivedN()
```

6. Iterate over the results and read each `PurchaseInfo` object.

# Sample Application: "InAppBuying"

- Open the project in  
`\<BADA_SDK_HOME>\Samples\InAppBuying\src.`
  - `InAppBuying.cpp`
- The InAppBuying application shows how to:
  - Retrieve item information from the Samsung Apps server.
  - Purchase an item.
  - Retrieve purchase information from the Samsung Apps server.



# FAQ (1/2)

Do I need my customer's credit card information to make a sale?

No. You never know, and never need to know, any customer credit card information. Everything is taken care of for you. Just sit back, and wait for the money to roll in.

- Buyers register their credit card information at Samsung Apps when they sign up.
- The Samsung Apps servers then maintain the buyer's credit card information linked to the buyer's ID.
- When buyers input their ID and password, Samsung Apps automatically finds their credit card information and proceeds with the purchase.



# FAQ (2/2)

Does the actual payment get processed when I call the `PurchaseService::PurchaseItem()` method in the bada Simulator?

No, the payment process runs in test mode. Payments are not actually processed until the application has been approved by Samsung Apps Seller Office. When the application has been approved by Samsung Apps Seller Office, the actual payment is processed both in the bada Simulator and on the target device as long as the application is available in Samsung Apps. The application becomes unavailable when Samsung Apps sets the application status to “retired” or “terminated”.



# Review

1. In order to make a sale, you need 2 critical pieces of information. What are they?
2. What method must you call to make a sale?
3. What parts of a customer's credit card information do you need to know?
4. How can you sell licenses for your application and display the customer's name in your "About" box?
5. Where does your application's list of `ItemInfo` objects originate from?



# Answers

1. You must know your item group ID that you received in the Samsung Apps Seller Office and the item ID for the item you want to sell.
2. Call `PurchaseService::PurchaseItem()`.
3. You do not need to know any of it.
4. The Store namespace does not have any provisions for customer information like that. You would need to roll your own solution.
5. All item information originates with you. You must enter it into Samsung Apps Seller Office for approval prior to it being accepted and going live.





**bada**

The platform with more opportunities  
Invitation to Adventure