IFML by Example:  
Modeling an Online Bookstore

This document exemplifies the versatility and adaptability of IFML by modeling the most common features available in a simple UI for a point of sale (POS) management, specifically targeted to a bookstore environment.
1 Content Model

During the session, an User is assigned a Shopping cart that at the beginning is empty. As the user browses through the page and gets information about the products available, adds products to the shopping cart. The list of products selected at the moment by the user, can be consulted at any time, offering the option of pay the current order, empty the cart or continue browsing in order to add more products.

![Diagram of Content Model of the Online Bookstore](image)

Figure 1: Content Model of the Online Bookstore
2 Process Model

When the user enters into the website, starts exploring the products available. Once he finds a product of interest, selects it, and the item goes to the shopping cart. The user can either keep exploring products in order to add more items to his order, or continue to manage the shopping cart by deleting all the products, or updating quantities of the selected ones. Once the user is ready to proceed with the payment, performs the checkout.

In order to authorize the payment, it's necessary to send the customer information to the bank entity, and wait for the confirmation. This procedure is illustrated in the Figure 2.

![Figure 2: Process Model of the Online Bookstore](image)
3 Model of the User Interaction Flow

Figure 3 shows the home page of the online Bookstore. In this section, the user can select one of the product categories, or go directly to the shopping cart.

After selecting a category, a list of products is displayed. For instance, Figure 4 shows all the products belonging to the books category.
When the user selects a product obtains the details of the selected item (such as full description and price) along with the option to add the product to the cart, as shown in the Figure 5.

The procedure described in the figures 3, 4, and 5 is represented in IFML as shown in the Figure 6. Once the user selects a category from CategoryList a navigation event is produced, and as a result, the products corresponding to the SelectedCategory are displayed. Similarly, when the user selects a product from ProductList, the details of the SelectedProduct are displayed.
When the user decides to buy the product and add it to the cart, causes a modal view container to be displayed, where the user must provide the quantity of items of the desired product (see Figure 7). After accepting the quantity, the article is added to the cart, and a confirmation window is displayed as shown in the Figure 8.

Figure 6: IFML model corresponding to the exploration of products

When the user decides to buy the product and add it to the cart, causes a modal view container to be displayed, where the user must provide the quantity of items of the desired product (see Figure 7). After accepting the quantity, the article is added to the cart, and a confirmation window is displayed as shown in the Figure 8.

Figure 7: Window displayed in order to catch the number of items desired by the user

Figure 8: Confirmation window for the action add to cart

Figure 9 shows the model fragment that adds a product to the cart: once the user press the add to cart button, a modal window appears asking for the quantity of items desired. This value, along with the SelectedProduct are submitted as parameters and represent the input of the add to cart action triggered. Once the action is performed, a confirmation window is displayed.
The shopping cart is the list of products previously selected by the user. In this section are shown the quantities and the order details. The user is able to update the cart by changing the quantities, empty the cart by deleting all the products of the current order, and start the payment process by clicking in the checkout button (see Figure 10).

When the user chooses to update the cart, the total amount is recalculated.

When the user empties the cart is redirected to a confirmation page as shown in the next Figure 11.
As illustrated in the IFML model of the Figure 12, when the user decide to delete all the items from the current order, the action *Empty the cart* is triggered, and after its execution, a confirmation message is displayed.

In the *Update* event, the user modify the values of the quantities and submits them by clicking in the button *Update*; this event causes an *Update* action to be triggered after which the shopping cart is redisplayed (see Figure 12).

Once the user has decided to perform the payment, he must provide his personal information and press “Next” (see Figure 13).
After providing his personal information, the user must provide his bank account information and confirm the payment in order to proceed with the transaction (see Figure 14). After performing the transaction, a confirmation page appears showing the details of the payment as shown in the Figure 15.

Copyright © 201 © www.ifml.org and webratio.com
When the user chooses the Checkout option, the container *Customer Information* is displayed. The user must provide his personal information by filling out the form within this container.

After the user submits his personal information, the container *Payment Information* is displayed. In this container the user must provide his bank account details. The name of the user (sent previously as the...
parameter: Name), is forwarded along with the credit card number (CC) and the total amount of the offer (previously sent by the shopping cart container) to the payment action (Execute the payment).

![Figure 16: IFML model corresponding to the event Checkout](image)

After the payment execution, a confirmation message is displayed with the transaction details. The IFML representation of this procedure is shown in the Figure 16.

To increase reusability and modularization in the models, designers may decide to cluster homogeneous parts of the model into Modules. For instance, the part of the model that deals with the payment management can be packaged into a specific module. This would simplify the model of the application, which would appear as in Figure 17.

The definition of the corresponding module is shown in Figure 18.
Figure 17: IFML Module Representation of the Checkout Event

Figure 18: Inner Process of the Module Payment Execution
4 System Modeling

IFML can be suitably used together with UML models and other OMG standards (e.g., BPM models).

For instance, UML sequence diagrams complement IFML models at the purpose of highlighting sequences of activation of client- and server-side components depending on user interaction events.

In the example, when the customer chooses the option update, the Browser sends a message to the WebServer with the id of the product and the new quantity, then the WebServer updates the shopping cart and returns a confirmation message.

If the user decides to delete all the products previously selected, he clicks the empty cart button, sending the message to the Browser. The Browser sends a message to the WebServer who is in charge of executing the deleting action and return a confirmation message.

When the user is ready to proceed with the payment, notifies the Browser who asks to the WebServer for the customer information form. After the WebServer returns the form, the Browser displays it. The next step to continue with the payment is wait for the user to fill out the form with his personal and bank information. When the user submits his information, this is sent to the WebServer who asks the DataServer to return the customer information in order to verify it. After verifying the customer information, the WebServer sends it to a ExternalBankService who is in charge of authorize the payment. Finally, after the WebServer receives the confirmation from the ExternalBankService, sends a confirmation message to the Browser.
Additional diagrams can be used to describe the deployment of the components and other aspects, as shown in Figure 20.
Figure 20: Deployment Diagram of the Online Bookstore