Second Workshop for Product Evaluation
My second process
# Table of Contents

**Workshop: Office Supply Request** .......................................................... 3  
Description of the Office Supply Request Process ........................................ 3  
Scope .............................................................................................................. 3  
Modeling the Process .................................................................................... 4  
1. **Model the process** .................................................................................. 4  
2. **Process data** ........................................................................................ 11  
    Define the Process Entity for the Sub-Process .......................................... 23  
3. **Forms creation** ..................................................................................... 24  
    Activity: Register office supply request.................................................... 26  
    Activity: Approve request ........................................................................ 36  
    Activity: Quotations – Request quotations ............................................. 39  
    Event: Receive quotations ...................................................................... 45  
    Activity: Select vendor ............................................................................ 49  
    Activity: Receive products ..................................................................... 53  
    Activity: Process invoice ....................................................................... 59  
4. **Business rules** ....................................................................................... 63  
    Transition Conditions ............................................................................. 63  
    Transition to Notify approval .................................................................. 64  
    Transition to Notify Rejection .................................................................. 67  
    Transition to end of Sub-Process ............................................................. 69  
    Transition to Select vendor ..................................................................... 70  
    Activity events ......................................................................................... 70  
    Set selected vendor ................................................................................. 75  
5. **Performers** .......................................................................................... 78
Activity: Register office supply request ................................................................. 79
Activity: Approve request ..................................................................................... 79
Activity: Receive products .................................................................................... 81
Activity: Process invoice ....................................................................................... 81
Activities for the Sub-Process .............................................................................. 83

6. Integrate with other services ........................................................................... 83
   Install the Web Service locally ........................................................................ 83
   Invoke Web service from Bizagi ....................................................................... 87

7. Work Portal ........................................................................................................ 91
   Users .................................................................................................................. 92
   Testing the Work Portal ..................................................................................... 93
   Optional mobile approval ................................................................................. 96

8. Sending notifications (optional) ................................................................. 103
   Activity: Notify changes required ................................................................. 103
   Activity: Notify Rejection ............................................................................... 104
   Activity: Notify Approval ............................................................................... 104

9. Conclusion ........................................................................................................ 105
   Next steps ....................................................................................................... 105

10. Troubleshooting .............................................................................................. 105
Workshop: Office Supply Request

The primary purpose of this exercise is to cover features that were not included in the first workshop, which are more advanced and equally important, such as Collections (one to many relationships), Tables in Forms, creation of a Sub-Processes, the creation of the Data Model from the Forms Designer among others.

The first workshop for product evaluation is a prerequisite for this workshop, since some subjects that were covered in the first workshop will not be explained again.

We will design an Office Supply Request Process. The information about the Process and data to be controlled and visualized can be found in the following pages. Follow step by step the instructions of this manual and at the end you will have an automated Process of the Office Supply Request.

Description of the Office Supply Request Process

The Office Supply Request Process starts when an employee of the organization submits an office supply request.

Once the requirement is registered, the request is received by the immediate supervisor; the supervisor must approve, request changes or reject the request. If the request is rejected the Process will end. If the request requires changes it is returned to the employee (requester) who can review the comments and change it.

If the request is approved it will go to the Purchase Department in charge of quoting and selecting a vendor. If the vendor is not valid in the system the Purchase Department will have to choose a different vendor.

After a vendor is selected and confirmed, the system will wait for the products to be delivered and the invoice received. The system will then send a notification to the employee informing the outcome.

Scope

This Process focuses in controlling the requests of Office supplies of the organization, from the initial request up to their delivery.
The Quotations Sub-Process will be defined as follows:

**Modeling the Process**

We’ll use Bizagi Studio to model and automate the Process. Click the Bizagi icon on your desktop to open the application.

1. **Model the process**

   Process modeling is the first step in the automation of a Process. Bizagi Process Modeler is a business process modeling and documentation tool. The modeler enables you to visually diagram, model and document business processes in industry-standard BPMN (Business Process Model and Notation). BPMN is a worldwide accepted format for process modeling.

   Your company is planning to create this Process in the same Bizagi Application; the new Process should be created together with the “Vacations Request” Bizagi project.
Open Bizagi Studio and select the **BPMProject** project. To create a Process click the **New Process** link from the first step of the Process wizard.

Type the name of the Process, **Office Supply Request**.

The Process Modeler, included in the Bizagi BPM Suite, will open.

Model the Process diagram. Modify the name of the process. Include functional areas and rename them. Your model should look like the image below.
Diagram the following Process:

The three *Notifications* activities must be transformed to Script tasks since these are activities executed automatically by the system. Right-click on them and change their type to *Script task*. 
A Link is a mechanism for connecting two sections of a Process. Link Events can be used to create looping situations or to avoid long Sequence Flow lines. In this Process we will use the link events to connect the *Receive Product* activity to the *Process Invoice* activity.

Transform the *Received* events to Link Events. Make sure these events has exactly the same name.

[Click for more information about Link Events.](#)

Convert the *Quotations* Activity to a Sub-Process. Right-click on it and transform it to a Sub-Process:
The Sub-Process wizard will display. Select the *embedded* option; this Sub-Process will only be used for the OfficeSupply Request.

Click on **Next** and enter the name for the Sub-Process: *Quotations*. Click **Finish**.
Save and close your Process.

So far your Process should look like this:

We are now going to design the *Quotations Sub-Process*.

Right-click on the *Quotations* Sub-Process and select *Edit subprocess* option:
A new Model window will display for the Sub-Process. Diagram the following Process:

The **Verify Vendor** Task will be an interface. Define this Task as a Service Task. Go to the Task’s properties and define it as asynchronous.

[Click here for more information about Asynchronous Activities.](#)
Some parameters (properties) like the Duration and Help text must be updated in the Process Activities when the Process diagram is completed. Set those properties in the Processes Activities as follows:

<table>
<thead>
<tr>
<th>Activity</th>
<th>Help Text</th>
<th>Duration</th>
</tr>
</thead>
<tbody>
<tr>
<td>Register office Supply Request</td>
<td>Register the information of your request</td>
<td>2</td>
</tr>
<tr>
<td>Approve request</td>
<td>Approve or reject the products requested</td>
<td>3</td>
</tr>
<tr>
<td>Receive products</td>
<td>Mark the products as received and enter the Invoice</td>
<td>2</td>
</tr>
<tr>
<td>Process invoice</td>
<td>Make the internal process for the invoice to be paid</td>
<td>5</td>
</tr>
<tr>
<td>Request quotations</td>
<td>Generate and print the Quotations to send to the vendor</td>
<td>3</td>
</tr>
<tr>
<td>Select vendor</td>
<td>Generate and print the Quotations to send to the vendor</td>
<td>3</td>
</tr>
</tbody>
</table>

When you are finished, save and close the diagram modeler windows to return to the Process Wizard.

2. Process data

Once the Process Diagram has been created we’ll proceed to the Data Model creation. This Data Model will include all the information required by the Process.

A structured data model will be used to represent the information. It may happen that we don’t know exactly what attributes will be used in the automation; we will create the basic core attributes and add more attributes later in the design.

1. In the second step of the Process Wizard select Model Data.
2. A window will display where you should enter the Process Entity. The Process Entity is the entity that gives you access to the rest of the data model, the starting point. Name it Office Supply Request and click **Ok**.

3. Create the Master Entities: Product Request, Quotations and Vendor Request. To add a Master Entity click the button located in the Entity Bar called **Master**. Drag and drop it and type the name, in this case Product Request.
Repeat the procedure for the *Quotations* and *Vendor Request* entities:
4. Create the Parameter Entities *Product Type*, *Vendor* and *City*. Drag and drop the Parameter button from the Entity Bar and type the name of the entity: *Product Type*. Repeat the procedure for the *City* and *Vendor* entities.

You may need to reorganize the Entities to make them look as you wish.
Right-click on *Office Supply Request* and edit its attribute list.

Include the following attributes:

<table>
<thead>
<tr>
<th>Display Name</th>
<th>Type</th>
<th>Length</th>
</tr>
</thead>
<tbody>
<tr>
<td>Request date</td>
<td>Date – Time</td>
<td>-</td>
</tr>
<tr>
<td>Request justification</td>
<td>String</td>
<td>100</td>
</tr>
<tr>
<td>Delivery date</td>
<td>Date – Time</td>
<td>-</td>
</tr>
<tr>
<td>Order total</td>
<td>Currency</td>
<td>-</td>
</tr>
<tr>
<td>Quotation request</td>
<td>File</td>
<td>-</td>
</tr>
<tr>
<td>Invoice</td>
<td>File</td>
<td>-</td>
</tr>
<tr>
<td>Invoice total</td>
<td>Currency</td>
<td>-</td>
</tr>
<tr>
<td>Invoice address</td>
<td>String</td>
<td>50</td>
</tr>
<tr>
<td>Invoice number</td>
<td>String</td>
<td>20</td>
</tr>
<tr>
<td>Invoice approved</td>
<td>Boolean</td>
<td>-</td>
</tr>
<tr>
<td>Products received</td>
<td>Boolean</td>
<td>-</td>
</tr>
<tr>
<td>Products received date</td>
<td>Date – Time</td>
<td>-</td>
</tr>
</tbody>
</table>

The result must look like the following image:
Click on Finish.

5. Create the attributes related to the \textit{Product Request} entity as follows:

<table>
<thead>
<tr>
<th>Display Name</th>
<th>Type</th>
<th>Length</th>
</tr>
</thead>
<tbody>
<tr>
<td>Quantity</td>
<td>Integer</td>
<td>-</td>
</tr>
<tr>
<td>Approved</td>
<td>Boolean</td>
<td>-</td>
</tr>
<tr>
<td>Rejection comments</td>
<td>String</td>
<td>100</td>
</tr>
<tr>
<td>Total price</td>
<td>Currency</td>
<td>-</td>
</tr>
<tr>
<td>Unit Price</td>
<td>Currency</td>
<td>-</td>
</tr>
<tr>
<td>Comments</td>
<td>String</td>
<td>100</td>
</tr>
</tbody>
</table>

6. Create the attributes related to the \textit{Vendor Request} entity as follows:

<table>
<thead>
<tr>
<th>Display Name</th>
<th>Type</th>
<th>Length</th>
</tr>
</thead>
<tbody>
<tr>
<td>Selected</td>
<td>Boolean</td>
<td>-</td>
</tr>
</tbody>
</table>

7. Create the attributes related to the \textit{Quotations} entity as follows:
At this point your model should look like the following image:

<table>
<thead>
<tr>
<th>Display Name</th>
<th>Type</th>
<th>Length</th>
</tr>
</thead>
<tbody>
<tr>
<td>Quotation total</td>
<td>Currency</td>
<td>-</td>
</tr>
<tr>
<td>Quotation date</td>
<td>Date-Time</td>
<td>-</td>
</tr>
<tr>
<td>Quotation</td>
<td>File</td>
<td>-</td>
</tr>
<tr>
<td>Selected</td>
<td>Boolean</td>
<td>-</td>
</tr>
</tbody>
</table>

8. Create the attributes that represent the relationships between the Office Supply Request (Process Entity) and other entities, as follows. An Office supply request will have multiple products (ProductRequest) associated and can also have multiple Quotations. Thus, these relationships must be created as collections. To create the collection from Office Supply Request to Product Request, add a collection attribute as follows:
9. Repeat the procedure for the relationship between Office Supply Request and Quotations.

A request will be fulfilled by a single vendor and so is represented by a related entity relationship. Add a new related entity relationship for Vendor request.
The request will have a delivery city. In the master entity create a relationship to the parameter entity, Delivery city. Add a new related entity relationship for Delivery city.

The following table shows the relationships between the entities.

<table>
<thead>
<tr>
<th>From entity</th>
<th>To entity</th>
<th>Display name</th>
<th>Relation Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>Office Supply Request</td>
<td>ProductRequest</td>
<td>Product Request</td>
<td>Collection (1-n)</td>
</tr>
</tbody>
</table>
Click **Finish** to save the changes.

10. Include the attributes of the Parameter entities.

**Table: **Product Type

<table>
<thead>
<tr>
<th>Display Name</th>
<th>Type</th>
<th>Length</th>
</tr>
</thead>
<tbody>
<tr>
<td>Product type</td>
<td>String</td>
<td>50</td>
</tr>
</tbody>
</table>

**Table: **City

<table>
<thead>
<tr>
<th>Display Name</th>
<th>Type</th>
<th>Length</th>
</tr>
</thead>
<tbody>
<tr>
<td>City name</td>
<td>String</td>
<td>50</td>
</tr>
</tbody>
</table>

**Table: **Vendor

<table>
<thead>
<tr>
<th>Display Name</th>
<th>Type</th>
<th>Length</th>
</tr>
</thead>
<tbody>
<tr>
<td>Vendor name</td>
<td>String</td>
<td>50</td>
</tr>
<tr>
<td>Vendor address</td>
<td>String</td>
<td>100</td>
</tr>
<tr>
<td>Vendor telephone</td>
<td>String</td>
<td>20</td>
</tr>
<tr>
<td>Vendor email</td>
<td>String</td>
<td>30</td>
</tr>
</tbody>
</table>

11. Create the relationships between the entities:

<table>
<thead>
<tr>
<th>From entity</th>
<th>To entity</th>
<th>Display name</th>
<th>Relation Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>Office Supply Request (master)</td>
<td>Product type (param)</td>
<td>Product type</td>
<td>Related entity</td>
</tr>
<tr>
<td>Vendor (param)</td>
<td>City (param)</td>
<td>Vendor city</td>
<td>Related entity</td>
</tr>
<tr>
<td>Vendor request (master)</td>
<td>Vendor (param)</td>
<td>Vendor</td>
<td>Related entity</td>
</tr>
<tr>
<td>Quotations (master)</td>
<td>Vendor (param)</td>
<td>Vendor</td>
<td>Related entity</td>
</tr>
</tbody>
</table>
At this point, the model should look like this:

12. Add values to the Parameter Entities. At this point it is common not to know all the values that parameter entities will have. You can add some, for example for the Product Type entity and later we will see how to add values from other steps in the Process Wizard.

Right-click the Product Type Entity and select Values.

Click the Add... Producttype button to include values. The records identifier will be generated automatically. You only need to type the description.
Enter the following values:

<table>
<thead>
<tr>
<th>Product Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>Pen</td>
</tr>
<tr>
<td>Pencil</td>
</tr>
<tr>
<td>Notebook</td>
</tr>
<tr>
<td>Printer Paper</td>
</tr>
<tr>
<td>Mouse</td>
</tr>
<tr>
<td>Keyboard</td>
</tr>
</tbody>
</table>

Close the window when you are done.

Do the same for the rest of the parameter entities

<table>
<thead>
<tr>
<th>City</th>
</tr>
</thead>
<tbody>
<tr>
<td>New York</td>
</tr>
<tr>
<td>Paris</td>
</tr>
<tr>
<td>London</td>
</tr>
</tbody>
</table>
13. Close the diagram and save the changes made to the model.

**Define the Process Entity for the Sub-Process**

We designed the Sub-Process through the Parent Process, and created the Data Model for the Parent process. Now, we need to define the Process Entity for the Sub-Process to proceed with the next steps of the automation.

1. In the Process Wizard view, select *Quotations* from the Process drop-down list, on the upper right corner of the window.

2. Click on Next in order to enable the second step of the Process Wizard and select Model Data.

   In the new window select the Process Entity. In this case the Office Supply Request Process and the Quotations Process have the same Process Entity. Thus, select the *OfficeSupplyRequest* entity from the drop-down list.

<table>
<thead>
<tr>
<th>Vendor name</th>
<th>Vendor address</th>
<th>Vendor email</th>
<th>Vendor telephone</th>
<th>Vendor City</th>
</tr>
</thead>
<tbody>
<tr>
<td>Vendor 1</td>
<td>1234 Bizagi road</td>
<td><a href="mailto:Vendor1@test.com">Vendor1@test.com</a></td>
<td>1234</td>
<td>New York</td>
</tr>
<tr>
<td>Vendor 2</td>
<td>4567 BPM road</td>
<td><a href="mailto:Vendor2@test.com">Vendor2@test.com</a></td>
<td>4567</td>
<td>Paris</td>
</tr>
<tr>
<td>Vendor 3</td>
<td>7890 Model road</td>
<td><a href="mailto:Vendor3@test.com">Vendor3@test.com</a></td>
<td>7890</td>
<td>London</td>
</tr>
</tbody>
</table>
3. The Data Model diagram will display. Right-click on the entity and select Add Related Entities. This option will include in the diagram all the entities related to the Process Entity.

4. Arrange the entities as you desire. Save and close the diagram. Process data

3. Forms creation

Once the Process Diagram and the Data Model are ready we will proceed to create the forms (screens) associated with each one of the human activities of the Process.
Forms are used to enter and display required information, so that end users may interact with the Process.

Bizagi helps modern businesses join the mobile work revolution and takes business processes automation to the next level.

You define a unique user interface through our powerful forms designer. Bizagi does the rest and optimally displays the information for each specific device.

In the Process Wizard, go back to the **Office Supply Request Process** by selecting it in the upper right corner.

Now go to the Third step of the Process Wizard: **Define Forms**.
In the new window, the activities with no associated forms will be highlighted with an exclamation mark. Click on the desired activity to define its associated Form.

Below we are going to define the information to be displayed in each of the Process Activities.

**Activity: Register office supply request**

The first Activity of the Process, *Register office supply request*, must contain the following information:
1. Go to the third step of the Process Wizard and select Define Forms. You’ll see a diagram where only user Tasks are available in which to create Forms. User Tasks that have no forms associated will be highlighted with an exclamation mark. Select the Activity *Register office supply request* by clicking it.

The following screen will display:
2. Drag and drop three Groups from the Containers section in the Controls tab, and name them: Request information, Product information and Delivery information.

3. Go to the Data tab and expand the OfficeSupplyRequest entity. Drag and drop the attributes: Request date, Request justification and Order Total. When an attribute or control type element is added to a Form, it is interpreted by Bizagi.
and becomes a **Control**. In the Basic Properties, change the Request date Control to be read-only (not editable).

4. Go to the Layout tab. Drag and drop a 50%-50% layout to the first Group.

5. When in the Layout tab, you will be able to manipulate the layouts included in your Form. To be able to manipulate Controls and add more attributes to the Form go back to the Data tab.
Drag and drop *Request date* to the left half of the layout.

6. At this point we realize that we need to create two more attributes to include in the Forms that were not created in the Data Modeling step: the Employee, requesting the purchase, and the Delivery address. We can create them from the Forms Designer as follows:

Right-click on the *Office Supply Request* entity and select *Edit Entity*. The Entity wizard will display.
Click Next.

Add both attributes as follows:

<table>
<thead>
<tr>
<th>Display Name</th>
<th>Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>Employee</td>
<td>Related entity - System - WFUSER</td>
</tr>
<tr>
<td>Delivery Address</td>
<td>String (Length 50)</td>
</tr>
</tbody>
</table>
Click Finish. The Data Model will be automatically updated with the two new attributes.

7. Drag and drop the Employee attribute to the second half of the layout.

Click the Control and then click on the Gear icon that is displayed on its upper right corner. This gear will open the control's Properties. Locate the Display attribute property and select fullName.
8. Go to the Layout tab and drag and drop a 30% - 70% layout to the Delivery information Group. 
Select it and change the relation to 40% - 60% in the Properties tab.

9. Go back to the Data tab. Drag and drop Delivery city and Delivery date to the first section of the layout. Drag and drop Delivery address to the second one.
10. All the delivery controls should be required. Click on each one and change the `Required` property to true.

11. We will now include the products information. Drag and drop the `Product request` collection to the `Product information` Group. This will display a Table Control. Click it to enable the `Edit columns` button.
12. Click the **Edit columns** button. This will display the columns area of the table. A table control must include at least one column. Drag and drop the attributes that will make up the columns of the table. Alternatively, a simple double-click of the attribute will automatically include it as a column. Add the attributes and set their properties as given in the table below. Click the attribute to enable its properties in the panel on the left.

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Visible</th>
<th>Editable</th>
<th>Required</th>
</tr>
</thead>
<tbody>
<tr>
<td>Product Type</td>
<td>true</td>
<td>false</td>
<td>false</td>
</tr>
<tr>
<td>Quantity</td>
<td>true</td>
<td>false</td>
<td>false</td>
</tr>
<tr>
<td>Comments</td>
<td>true</td>
<td>false</td>
<td>false</td>
</tr>
</tbody>
</table>

Click **Ok**.

13. Tables in Bizagi can have separate self-contained forms to add records, or additions can be manipulated inline. We will allow this table to add records inline. Click on the table and select the Gear icon that is displayed on its upper right corner. Go to the **Advanced** tab and verify that the following properties are enabled: **Allow Delete**, **Allow Add** and **Inline add**.
Enable the *Allow Edit* and *Inline edit* property as well.

14. Save the Form and close it.

**Activity: Approve request**

In the third step of the Process Wizard select the activity Approve request. The final form should look as below:
1. The first part of the Form will be exactly the same as the one designed before, so we will use the Copy From option to help us design faster (for more information on this feature you may see the First workshop for Product Evaluation). Change all the controls to Not Editable, except the table.

2. We will edit the table to include the options for the Supervisor to approve or reject the products. Select the table, click on Edit columns, and convert the three existing columns to Not Editable.

3. Add two more columns: Approved and Rejection comments. Click Ok.
4. You will be taken back to the main form of the Approve request Activity. Select the table’s Gear icon to change its properties. Disable the Allow Add and Allow Delete properties to prevent the Supervisor from adding or deleting any records.

5. Save and close the Form.
Activity: Quotations – Request quotations

The next step is to create the first form of the Sub-Process. In this activity the Purchase department will generate a Quotation request. Bizagi will create the document based on a predefined template, so the department can send it via email or fax.

Download the template Quotation template.docx from:

http://download.bizagi.com/myfirstandsecondprocess/mysecondprocess/MySecondProcess.zip

Save this file in a local folder for easy access.

1. Click on the Quotations Sub-Process. A new window for the Sub-Process will display. Click on the Request quotations Activity.

2. Include the following information

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Visible</th>
<th>Editable</th>
<th>Required</th>
<th>Group</th>
</tr>
</thead>
<tbody>
<tr>
<td>Request Date</td>
<td>true</td>
<td>true</td>
<td>false</td>
<td>Request Information</td>
</tr>
<tr>
<td>Employee (fullName)</td>
<td>true</td>
<td>false</td>
<td>false</td>
<td>Request Information</td>
</tr>
<tr>
<td>Delivery address</td>
<td>true</td>
<td>false</td>
<td>false</td>
<td>Delivery Information</td>
</tr>
<tr>
<td>Delivery city</td>
<td>true</td>
<td>false</td>
<td>false</td>
<td>Delivery Information</td>
</tr>
<tr>
<td>Delivery date</td>
<td>true</td>
<td>false</td>
<td>false</td>
<td>Delivery Information</td>
</tr>
<tr>
<td>Products request</td>
<td>true</td>
<td>false</td>
<td>false</td>
<td>Quotation information</td>
</tr>
</tbody>
</table>

Click on the Products request table and then, click on Edit button to include the columns to be shown. Add the columns Product Type, Quantity and Comments.

3. As mentioned above the Quotation request document will be generated using Bizagi’s Document template functionality.

Go to the Controls tab and drag and drop a Document template from the Bizagi controls group, to the Quotation information Group.
4. Click on the control's Gear icon to display its properties.

- In the Button caption type: Generate Quotation request.
- In the Data source property select: OfficeSupplyRequest – Quotation request.
- Set the Required property to True.
5. Go to the Format tab and change the Display type to Value. This way only the button will display, with no label.

6. Return to the Basic tab of the control to configure the Template. Click the **Configuration Wizard** button.
   The wizard will display. Click the Add icon (plus sign) in front of the Template option.
7. Browse to the location where you previously saved the **Quotation template.docx** and select it.
   Click **Next**. Bizagi will display the related template. Then, click **Finish**.

8. Change the Document name to: **Quotation request**. This is name that Bizagi will display for the generated document in the Work Portal.
9. Click **Add mapping** to map the attributes from the Data Model to the Template.
On the left, you will find the attributes in Bizagi’s data model (Bizagi Data). On the right, you will find all the tags found on the template. Connect them using your mouse, each Tag in the document (on the right), to the corresponding attribute in Bizagi, as follows.

![Image of mapping diagram]

10. For the **QuotationDate** Tag click the Maximize icon on the upper right corner.

![Image of mapped template]

11. Select the **Advanced Tools** icon and drag and drop the **Today Date** function.
12. You will be asked to define the Date Format; select short date.

13. Then connect the Date to the *QuotationDate* Tag on the left.

14. Minimize the window and click **Finish**. When the mapping window closes, click **Ok** to close the Configuration wizard.

15. Save the Form and close it.
   Please refer to **Document templates** for more information.
Event: Receive quotations

Click on the Event *Receive quotations* in the third step of the Process. Once again, use the *Copy from* option and copy the information from the *Request quotations* Activity.

The Form will look as follows:

1. Delete the Products request table clicking the **Delete** icon on the upper right corner.
2. Hide the Document template Generate button so the end user of this Event will be able to view the file but not change it. Select the control, and in the **Advanced** tab disable the Display generate button property.
3. Return to the Data tab and drag and drop the *Quotations* collection with the following columns:

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Visible</th>
<th>Editable</th>
<th>Required</th>
</tr>
</thead>
<tbody>
<tr>
<td>Vendor (display attribute: Vendor Name)</td>
<td>true</td>
<td>false</td>
<td>false</td>
</tr>
<tr>
<td>Quotation Date</td>
<td>true</td>
<td>false</td>
<td>false</td>
</tr>
<tr>
<td>Quotation Total</td>
<td>true</td>
<td>false</td>
<td>false</td>
</tr>
<tr>
<td>Quotation</td>
<td>true</td>
<td>false</td>
<td>false</td>
</tr>
</tbody>
</table>

Make sure they are all *Not Editable*.

4. In the collection’s properties go to the Advanced tab and define the following:
   - Allow Delete enabled
   - Allow Add enabled, with Inline Add disabled. This will open a property to select a Form (that will be defined in the next step)
   - Allow Edit enabled with Inline Edit disabled. This will open a property to select a Form (that will be defined in the next step)
5. Select the **Add form** drop-down list and click **New Form**. This will display a new Forms Designer window in which we will create a form for the end user to add the quotations that are received. Include the following information in the Form:

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Visible</th>
<th>Editable</th>
<th>Required</th>
</tr>
</thead>
<tbody>
<tr>
<td>Vendor (display attribute Vendor name)</td>
<td>true</td>
<td>true</td>
<td>true</td>
</tr>
<tr>
<td>Quotation Date</td>
<td>true</td>
<td>true</td>
<td>true</td>
</tr>
<tr>
<td>Quotation Total</td>
<td>true</td>
<td>true</td>
<td>true</td>
</tr>
<tr>
<td>Quotation</td>
<td>true</td>
<td>true</td>
<td>true</td>
</tr>
</tbody>
</table>

The result as follows:

6. Go to the **Properties** button on the Ribbon and give the Form a Display name: *Add quotations*.

Save and close the Form and you will be back at the main **Receive quotations** Form. The Form you just created will be automatically assigned as the **Add Form**. Choose the same form as the Edit form:
7. Save and close the *Receive quotations* Form.

**Activity: Select vendor**

The next step is the creation of the form for the Select vendor activity. The idea of this Activity is to select a Vendor based on the information included in the *Receive quotation* Event.
1. Click on the Activity Select Vendor, in the third step of the Process Wizard. Once again, use the Copy from option and copy the information from the Receive quotations Event.

2. Go to the Quotations collection and open its Properties by clicking the Gear icon. In the Advanced tab define the following:
   - Allow Delete disabled
   - Allow Add disabled
   - Allow Edit enabled with Inline Edit enabled.
3. Include one more column in the collection: the attribute **Selected**. Mark this control as Editable and Required. Make sure all other columns are NOT editable. Click **Ok** to return to the main Form.

4. We will include a validation to make sure that only **ONE** quotation is selected.
Click the Gear icon in the Quotations table and select the *Advanced* tab. Then, click the **Table validations** button.

5. Define the validation:

   Give the validation a name: *One quotation selected*.

   Add a Counter by clicking the **Add Counter** button: *SelectedTrue* where Selected is equal to *Yes*.

6. Add the validation:

   Where the Counter *SelectedTrue* is Different to **ONE** (one must be typed as the number 1).
7. Add an Error Message to display to the end user when the condition is met: You must select only ONE quotation.

This error message will display if the final user tries to select more than one quotation, or if none are selected. Click Ok to save the Validation.

8. Save and Close the form.

9. Close the Sub-Process to go back to the main process.

**Activity: Receive products**

At this point the vendor has been selected.
We will define the form for the Receive products Activity where all product must be received.

1. Click on the Activity *Receive products*, in the third step of the Process Wizard. And use the Copy from option and copy the information from the *Register office supply* request Activity.
   Change all the controls to Not editable.
   Drag and drop the attribute *Products received* to the *Product information* group, mark it as Required.

2. Go to the Products request table and disable the Editable property to make sure that the end user cannot add, delete or edit any record.
3. We will now add a validation to make sure all products are received before the process can continue.

Click the **Actions & Validations** button in the Ribbon, and Add a new Validation.
4. The validation will be:
If the Control Products received is NO or EMPTY, then throw an error message.
In the If section:
- Click on Add a condition.
- Select the control Products received and the operator is false.
- Click once again on Add a condition.
- Select the control Products received and the operator is empty.
- Change the and join to an or.
5. In the *Then* section:
   - Click the word Message. This will display a text box to type the error message.
   - Type: *All products must be received before the process can continue.*
   - Click **Ok** to save the Validation.
6. Save and close the Form.

**Activity: Process invoice**

This is the last activity of the Process where the processing of the invoice and its payment is performed.

   Use the *Copy from* option and copy the information from the *Register office supply request* Activity.
   Change all the controls to Not editable, except for the Products request table.

2. Move the Delivery information group from the bottom of the form and position it above the *Product information* group.
3. Edit the columns of the Products table. Remove the control Comments and add two additional controls.

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Visible</th>
<th>Editable</th>
<th>Required</th>
</tr>
</thead>
<tbody>
<tr>
<td>Unit Price</td>
<td>true</td>
<td>true</td>
<td>true</td>
</tr>
<tr>
<td>Total Price</td>
<td>true</td>
<td>true</td>
<td>true</td>
</tr>
</tbody>
</table>
4. Change the Advanced properties of the table:
- Allow Delete disabled
- Allow Add disabled,
- Allow Edit enabled with Inline Edit enabled.
5. Add an additional Group, Vendor Information, below the Product information Group.
Include in the group the following information:

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Visible</th>
<th>Editable</th>
<th>Required</th>
</tr>
</thead>
<tbody>
<tr>
<td>Vendor - Vendor Name</td>
<td>true</td>
<td>false</td>
<td>false</td>
</tr>
<tr>
<td>Vendor - Vendor Email</td>
<td>true</td>
<td>false</td>
<td>false</td>
</tr>
<tr>
<td>Invoice address</td>
<td>true</td>
<td>true</td>
<td>true</td>
</tr>
<tr>
<td>Invoice number</td>
<td>true</td>
<td>true</td>
<td>true</td>
</tr>
<tr>
<td>Invoice total</td>
<td>true</td>
<td>true</td>
<td>true</td>
</tr>
<tr>
<td>Invoice</td>
<td>true</td>
<td>true</td>
<td>true</td>
</tr>
<tr>
<td>Invoice approved</td>
<td>true</td>
<td>true</td>
<td>true</td>
</tr>
</tbody>
</table>

6. Save the Form and close it.
7. Close the Forms Wizard by clicking on the Back (green arrow) icon to return to the Process Wizard.

4. Business rules

The next step in the Process Wizard is the definition of business rules that control the Process routing. The first rules to be completed are known as transition rules. These rules evaluate conditions and decide where the Process flow is to continue. They return True or False and they are associated to Gateway shapes.

Transition Conditions

1. To create a Business Rule go to the fourth step of the Process Wizard and select Define Expressions.

2. The Rule editor will open highlighting the transitions that have no rule associated.
The Approval outcome gateway leads to three possible paths:

- If the request was approved, the requester will be notified by email and quotations will be requested.
- If the request was rejected, the requester will be notified about the reasons for the rejection.
- If the request was rejected, the flow returns to the Register Request activity in order to make the required changes.

Bellow we are going to describe how to configure each of these transitions.

**Transition to Notify approval**

An Office Supply Request is approved if ALL of the products requested were approved. In order to evaluate this we have to identify how many products were requested and how many were approved. If the number of approved products is equal to the number of requested products, the request is approved.

To configure this evaluation follow the next steps:

1. Select the transition named Approval outcome that reaches the Activity Notify Approval by clicking on the transition.

2. Select the option Based on the result of an expression. Press New.

3. In the Boolean Expression click the Custom XPaths button. Then click the Add custom XPath button on the bottom.
4. Select the *Productrequest* Collection and then select the *count* function.

5. Click the *Add filter condition* button, to add a condition. Build the condition: *Approved* is equal to- set a constant value (True). This will evaluate how many products were approved.
6. Click **Set filter** and verify the complete XPath is built. Then click **Ok**.

7. Add a second Custom XPath selecting the *Product Request* collection and the *count* function without any filter. This will evaluate how many products were requested.
   Click on **Ok** to save the Custom XPath.
8. Drag and drop both Custom XPaths to compare them to each other, using the *Is equal* operator.

9. Click OK to save the rule.

**Transition to Notify Rejection**

The Process will follow this path when no products have been approved. In order to evaluate this we have to identify how many products were approved. If the number of approved products is equal to 0, the request is approved.
To configure this evaluation follow the next steps:

1. Select the transition named *Rejected* that reaches the Activity *Notify Rejection* by clicking on the transition. Select the option *Based on the result of an expression*.

2. In the Boolean Expression repeat the steps described for the previous transition from 1 to 4 and compare the Custom XPath to 0 (zero).

3. Click **Finish** to save the Boolean Expression.

**Transition to Notify Change Required**

This path will be followed when the other two conditions are NOT met. For this path we will select the *Else* rule. Select the transition from the Gateway that reaches the *Notify changes required* Task by clicking on it. Select the *Else* condition in the Expression Selection window and then click **Ok**.
Transition to end of Sub-Process

The status of the attribute *Selected* located in the Vendor entity will be the result of the *Office Supply Request* web service that will evaluate if the Vendor selected complies with the company's rules. If the vendor complies, the Sub-Process will end, if not, the flow must return to the *Select vendor* Activity.

1. Click on the *Quotations* Sub-Process will display highlighting the transitions that have no rule associated.

2. Click on the transition named *Yes* that reaches the end of the Sub-Process. In the Boolean Expression editor drag and drop the attribute *Selected* from the Data Model on the left to the condition item. Select the function *is Equal* to and choose the *true* option:
3. Click **Finish** to save the expression.

**Transition to Select vendor**

1. Click on the transition named **No** that reaches the Activity **Select vendor**.

2. Select the **Else** condition in the Expression Selection window and then click **Ok**.

3. Return to the Process Wizard by clicking the Back (green arrow) icon.

**Activity events**

**Set Employee and Date**

We will create rules to automatically fill-in the fields **Request date** and **Employee**; this way, when a case is started these two Controls will be filled by Bizagi containing the information of today's date and the Employee logged in.

1. As in the first workshop these rules will be created in the first Activity of the Process; we will include an Expression that sets the **Request date** and to save the Case Creator as the **Employee**:
   Go to the fourth step of the Process Wizard and select Activities Actions.
2. In the first activity select the **On Enter** option. Then click on the Plus icon to add an Action in the Activity and select **Expression**.

3. Select **New** to create a new expression.
4. The Expression editor will display. Fill the Display Name and Description as follows; this information will let you identify the Expression for later use.

5. Right-click on the black vertical arrow and select **Add Expression**. Type: Employee and date and click **Ok**.

![](image1.png)

![](image2.png)
6. Open the Expression editor by double clicking on it and add the assignments as we did in the first workshop for the Employee:
   - Select the Employee attribute from the Data Model. Then type ‘=’
   - Select the function User ID of case creator, found in the Case creator user Category.
7. To set the Date of the request:
   - Select the Request date from the Data model. Then type ‘=’.
   - Select the function **Today** from the **Date & time** Category.

8. Click Ok when done. Click Ok once more to save the Expression.
Set selected vendor

After a quotation is selected we need to take the Vendor and set it directly to the Office Supply Request, so the web service will be able to check the Vendor directly.

1. Click the Quotations Sub-Process. Click the Select Vendor Activity. Add an Expression to the On Exit option.

2. Name the expression SetVendor.

3. Add an Expression element with description: Set Vendor

4. Select from the Data Model the Vendor XPath.

5. Click OK and type ‘=’.

6. Select the Data Model once more to select the collection XPath.
   - Select the Quotations collection.
- Click Add filter condition button.
- Add the condition: *Selected equals to True*.

7. Click the **Set filter** button and verify the XPath is as follows:

8. Navigate to the **Vendor** attribute in the Quotations collection and then select the Vendor. This will return the selected Vendor in the collection. Make sure the XPath is as follows. Click **OK**.
9. The final Expression should be:

\[ \langle \text{OfficeSupplyRequest.Vendor.Vendor} \rangle = \langle \text{OfficeSupplyRequest.Quotations[Selected = true].Vendor} \rangle \]
5. Performers

The resources assignment is a very important stage within Bizagi. The responsible resources for each one of the Process activities are defined in this stage.

For this process we will have three different allocations:

- The Activities **Register office supply request** and **Receive products** will always be executed by the case creator, or Employee.
- The **Approve request** Activity will always be executed by the case creator’s supervisor.
- The Activities in the Purchase Department will be always executed by a person with the Position **Purchase Department Assistant**.

To allocate these performers we will use the same procedure as in the first workshop.

Click on the fifth step of the Process Wizard and select the **Define Performers** option. The activities that do not have a Performer configured will be the ones highlighted in the workflow.
Activity: Register office supply request

This Activity must be assigned to the creator of the case; this is a default functionality Bizagi provides, so despite the activity is highlighted with an exclamation mark, we don’t have to create an assignment for it.

Activity: Approve request

1. Once in the diagram select the Activity by clicking on it, you’ll find the Performers assignment window.

2. Click on **Add condition**.

3. In the new window we will select the Requester’s Boss as the performer. Select **User Id** in the first drop down list.

4. Click on **Select Expression** link and don’t save any change to the Expression.
5. Select the `CurrentAssigneeBoss` and click **Ok**. This instruction will automatically assign the creator’s Supervisor to the second task.

The final expression will be like this:
Activity: Receive products

Select the User Id property that equals the expression \textit{Case Creator} (Which is found in the Default expressions).

Activity: Process invoice

1. Select \textit{Process invoice} Activity and add a condition.
   Since we have no Positions created we will create one:
   In the first drop-down list, select \textit{Positions}.
   A new drop-down list is enabled at the far right. Click on \textit{Organization}. A link will appear to the right of the field.
2. Click **New** to add a new position. Type: *Purchase Department Assistant*. Click **Save**.
   This will automatically assign the property.

3. Click **OK** on the condition and once more on the Performers window to save the allocation rule.
Activities for the Sub-Process

Repeat the procedure of the Positions for all the activities in the Quotations Sub-Process by selecting the position Purchase Department Assistant we just created from the Value field in the Performer condition dialog.

Close the Diagram by clicking the Back (green arrow) icon located on the top right corner of the screen to return to the Process Wizard.

6. Integrate with other services

This second workshop includes an interface to verify the selected vendor information.

To verify the vendor all the information will be sent to the web service that will determine if the vendor complies with the company’s policies.

There are two options to be able to implement this step.

- The simplest one is if you have an internet connection. In this case you can use an existing web service available. No further configuration is needed and you can jump directly to the section Invoke the Web Service from Bizagi lower down.
- If you don’t have access to an internet connection install the web services locally, as described below.

Install the Web Service locally

Make sure you are connected to the internet. Download and unzip the file OfficeSupplyWS.zip that is found in:
http://download.bizagi.com/myfirstandsecondprocess/mysecondprocess/MySecondProcess.zip

1. Copy the unzipped folder into your default web site local path, usually C:\inetpub\wwwroot.

2. Open the IIS Manager either by running the command `inetmgr` or by selecting it form the Administrative Tools in the Control Panel.
3. WINDOWS 7 and 8
In the manager, right-click on the folder **OfficeSupplyWS** and select **Convert to Application**. Then click Ok.

WINDOWS XP
Open the Default Web Site and find the **OfficeSupplyWS** folder. Right-click on it and select **Properties**.
4. In the OfficeSupplyWS Properties window click the **Create** button and then **OK**.
Invoke Web service from Bizagi

Activity: Verify Vendor

1. From the Process Wizard go to the sixth step and select *Define Integration Interfaces*.

2. A view of the Process is presented. In this view, only the service Tasks will be active. Click on the Task *Verify Vendor* in the Sub-Process *Quotations*.

3. The Interface Wizard will display. First type the URL for the web service.
   - If the web service is installed locally, type: [http://localhost/OfficeSupplyWS/OfficeService.asmx](http://localhost/OfficeSupplyWS/OfficeService.asmx)
4. Click the Go button to see the available methods. Select the method `VerifyVendor` and click Next.

5. In this step the information from Bizagi to the web service will be configured. Two tables are shown, on the left is Bizagi’s Data Model and on the right the information the web service method expects. Map the information of Bizagi in `OfficeSupplyRequest – Vendor – Vendor` with the matching information in the web service. Then click Next to continue.
6. If no error is returned, the value of VENDOR_STATUS from the web service response is going to be saved in the Selected attribute of OfficeSupplyRequest – Vendor in Bizagi. Connect these two elements and click Next.

7. On the final step we will configure what to do if an error occurs.
For this case, the web service response message includes an error structure. The error value is the XPath: `IntVerifyVendorResponse/ERROR_MESSAGE/eCode`
If this code is different from zero (0) an error occurred in the web service. Then, the way to indicate this to the process is setting it in the Response errors section. Click the **Response XPath** output parameter to check for an error value and type the XPath: `IntVerifyVendorResponse/ERROR_MESSAGE/eCode` and click **Enter**.

8. Click the **Add error validation** link.
   Configure the validation as shown in the picture below.
   The Error message column can be a constant string or an XPath. For this exercise use the XPath: `IntVerifyVendorResponse/ERROR_MESSAGE/eMessage` for the error message.
9. Click **Finish**. When the interface wizard closes, return to the Process Wizard by clicking on the Back icon (green arrow).

The service task will not validate the vendor if the Name, Address, Telephone and City fields are blank.

Therefore, it is advisable to review the City and Vendor parameter entity values to ensure all these fields are set. Check out the [Troubleshooting](#) section to learn how to enter values from the Modules view (advanced).

### 7. Work Portal

So far, the Process flow has been completed along with the structuring of the Process information, the Forms design for each one of the Process activities, the definition of the business rules and the performers. Now we are ready to configure the Work Portal.

Click on the seventh step of the Process Wizard. In the new window select **Run Process** and then **Development**.
Users

The Work Portal has three users, so that the behavior of the assignments can be verified. The following table presents the user list. (Remember you have already created the Supervisor and Employee users).

If you need more information on User creation please visit our wiki or check the First Workshop for Product Evaluation.

<table>
<thead>
<tr>
<th>User</th>
<th>Password</th>
<th>Domain</th>
<th>Immediate Supervisor</th>
<th>Job Title</th>
</tr>
</thead>
<tbody>
<tr>
<td>Supervisor</td>
<td>Supervisor</td>
<td>domain</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Employee</td>
<td>Employee</td>
<td>domain</td>
<td>Supervisor</td>
<td></td>
</tr>
<tr>
<td>Purchase</td>
<td>Purchase</td>
<td>domain</td>
<td></td>
<td>Purchase Department Assistant</td>
</tr>
</tbody>
</table>

We will configure the Quick login functionality so you can test the Work Portal with all users without the need to type the complete name and password.

Open Bizagi Studio and click on the Modules icon located on the upper left corner. This will enable the Modules view, which is an advanced view.
Go to Security module and locate Enable Quick Login as shown in the image below.

Select ON (the tick mark should show) and click Update.

When Quick Login is enabled there is no need to enter the user's password in the Work Portal, simply select the user you want to log in as. Click Run to test the Work Portal.

**Testing the Work Portal**

1. Open the Work Portal, and login with the Employee user.
2. Create a new case for the **Office Supply Request** process and test all the activities.

![Image of Bizagi workflow](image)

3. Enter the Request justification and the Order total.

4. To add records to the table click the **Add icon** (plus sign) at the bottom left of the table. Enter the information and then click the **Save** button of the table.

![Image of product request table](image)

Complete the rest of the information and click **Next**.

5. Login with the **Supervisor** user. Enter the **Approve request** Activity.
   Approve at least one product to continue with the normal flow of the Process.
Optional mobile approval

You can also download our Bizagi mobile application and login to see how the Supervisor can approve a request from a mobile device. Your mobile device and your project must share the same network. That is, be connected to the same wi-fi network.

Your browser has an address. This is the one you need to enter the application.

Log in with the user Supervisor and go to the Approve request activity.
6. If you rejected a product the process will go back to the Register office supply request Activity. Click Next. If you approved all the products, the Process flow will continue to the Activity Notify approval.

7. Login with the Purchase user. Enter the Request quotations Activity. Click the Generate Quotations request button to generate the PDF document.
8. The document will be generated and the link inserted below the button. This might take a few minutes. Once done click the document name to open it.
9. When **Next** is clicked you will not have pending activities for the case. However an Event tab is enabled on the left to receive the quotations.

10. Enter the Event and add the Vendors that quoted for the products. Click the **Add** icon (plus sign) on the bottom left of the table and fill in the required controls. Click **Save** to return to the main Form. Then, click **Next** to continue in the Process.
11. In the Select vendor Activity choose one vendor. Remember that if you choose more than one or you choose none, there will be a validation:

12. Once the Select vendor Activity has finished, the Sub-Process will terminate as well. Log in with the Employee user and open the Receive products Activity found in the Inbox.

Note that if you specify that no products were received a validation will display.
13. Select Yes in the Products received control and click Next. Login with the Purchase user and open the Process invoice Activity.
Fill in all the information. This is the last Activity of the Process, as you click Next the Process will terminate.
14. You can view the status of the closed case in the Details tab of the Case area.
Please review our First Workshop for Product Evaluation if you need more information on testing the Work Portal.

8. Sending notifications (optional)

In the First Workshop for product evaluation we learned how to configure the SMTP server and how to configure notifications. If you completed the last workshop you won’t need to re-configure the SMTP server and the Employee’s email. Otherwise, please follow the configuration steps in the previous workshop.

The following notifications will be configured:

**Activity: Notify changes required**

![Image of e-Mail Definition Window with configuration details]
Activity: Notify Rejection

Dear <OfficeSupplyRequest.Employee.fullName>
Your supervisor has rejected your office supply request.
Case number <CaseNumber>

Regards.

Activity: Notify Approval

Dear <OfficeSupplyRequest.Employee.fullName>
Your office supply request has been approved and it is now in the Purchase Department to be processed.

Regards.
9. Conclusion

In this workshop, you learned how to:

- Create a collection.
- Include a table in a form, and manage it in the Work Portal.
- Use link events.
- Make semi-advanced rules
- Create document templates with business information.
- Convert and configure a service Task to an Asynchronous service Task.
- Configure a controlled web service message error.
- Create a Sub-Process.
- Handle the forms, rules, events and performers associated to a Sub-Process.
- Create validations in a table.
- Add values to a parameter entity from the Forms Designer and the Modules view.
- Modify entities from the Form Designer.

Thank you for completing this second workshop, for Bizagi it has been a pleasure to show you why we are the most agile solution to automate Process.

We hope that your experience with Bizagi has been satisfactory; should you have any questions or comments, please feel free to contact us at: http://www.bizagi.com/en/contact-us

Next steps

- To learn more about Bizagi and BPMN please take our online courses at http://elearning.bizagi.com/

10. Troubleshooting

This section deals with the most frequent questions arising from this workshop exercise:

1. How to add values to the Parameter entities, other than the Data Model and Forms Designer steps.
Go to the **Modules** view in Bizagi Studio and locate the Entities Module.

Go to the Parameter entity (in this case City) and go to values.

Click on the **Add button** located on the bottom of the window and add as many records as required.