

---

# Introducing Visual Paradigm plug-in development

Visual Paradigm provides an interface for developers to integrate with Visual Paradigm. Developers can develop their plugins for what they want. In this section, we will introduce the structure of a plugin.

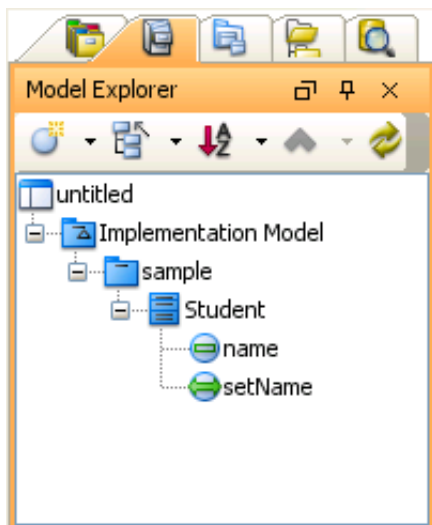
## Plugin.xml

A plugin is defined in a XML file (plugin.xml). It includes the information (such as plugin id, provider, required libraries, etc...), custom actions (menu, toolbar and popup menu) and custom shapes/connector of the plugin.

For working with Visual Paradigm in plugin, there are 4 main components must be known by developers: Model, Diagram, Diagram Element and Action/Action Controller.

## Model element

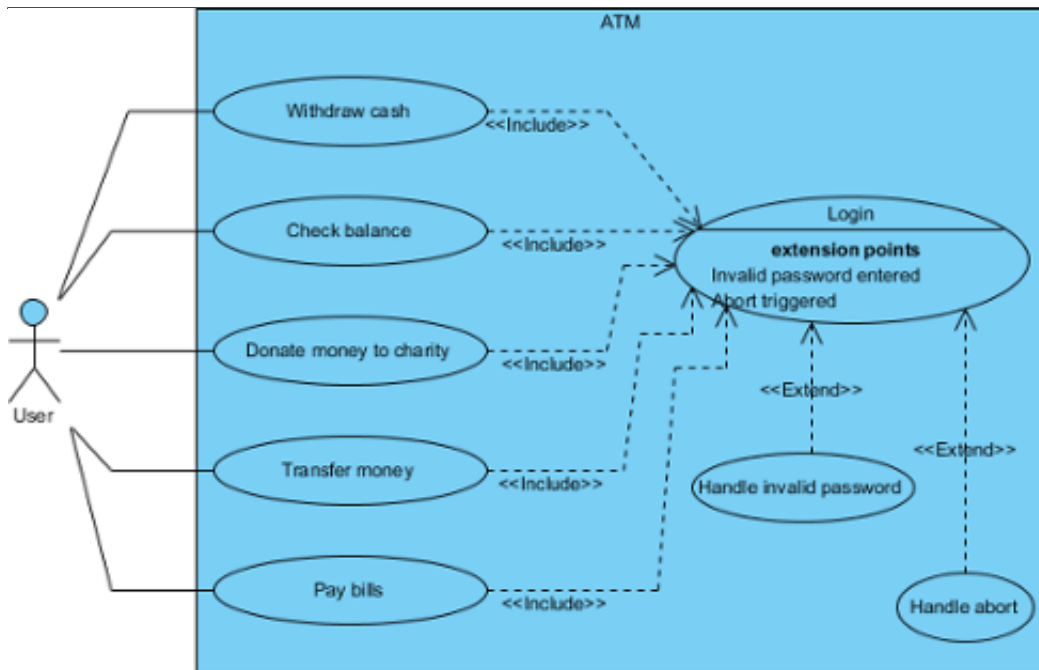
Model Elements are basic construct of a model. Plugin allows developer to create, retrieve, update and delete model elements through the popup menu context or through the project (by iterating model elements within a project).



## Diagram

Diagram is contains diagram elements on different domain (such as Use Case Diagram, Class Diagram, ERD, etc...).

Plugin allows developer to create, retrieve, update and delete diagrams through the popup menu context or through the project (by iterating diagrams within a project)



## Diagram element

A model element does not contain information of appearance (such as x, y, width, height, etc...). It is the diagram element, which appear on the user interface, that owns the appearance data. Diagram Element represents a view of a model element. A model element can be shown on different diagrams (such as a class can be shown on 2 different class diagrams).

There are 2 kinds of diagram element: Shape and Connector. Shape represents the non-relationships diagram element (such as Class). Connector represents the relationships (such as Generalization). Plugin allows developer to create, retrieve, update and delete diagram elements through the popup menu context or through the project (to iterate all the diagrams and then the diagram elements appear on a diagram).



## Action/Action controller

Action represents buttons and menus (menu, toolbar and popup menu), which contains the information on outlook (such as label, icon, mnemonic, etc...) and responses to trigger the function call.

Action is used to represent the button on 3 regions: menu/toolbar, popup menu and diagram toolbar

Action Controller is the control (function call) of actions. Developer needs to implement different Action Controller on different region's actions.

