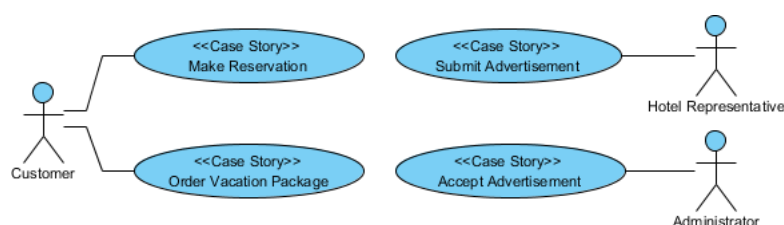




How to Write Effective Use Cases?

Written Date : January 27, 2016

One of the most difficult problem in software development is capturing precisely what you want to build. Inaccurate requirement will end-up with significant delay, rework or even abandonment of the project. Effective applying use case technique helps your team capturing requirements in user point of view which can be easily understood by both the end-user and your team. Use case driven development support subsequent development activities such as analysis and design and testing.



What is a Use Case?



Use case is something that the actors want to do for obtaining an observable business goals. They are named with a short verb or verb + noun phrase. You should use concrete and specific verbs and nouns to avoid ambiguity. Verbs like 'do' and 'perform' and nouns like 'data' and 'information' should be avoided whenever possible.

Theoretically, the end users will perform actions that are supported by the system to achieve their ultimate goals, as identified in use case analysis. Take online hotel reservation system as an example. "Make reservation" is undoubtedly a business goal, thus a use case. The function to look-up a hotel on an online map can also be what a user needs. However, it is not a use case because the action itself does not yield any observable goal.

It would not be appropriate to model requirements related to implementation issues as use cases, such as: support multiple look & feel, deployment arrangement, construct database. All these are wrong and could lead to bad, or even wrong system being built.

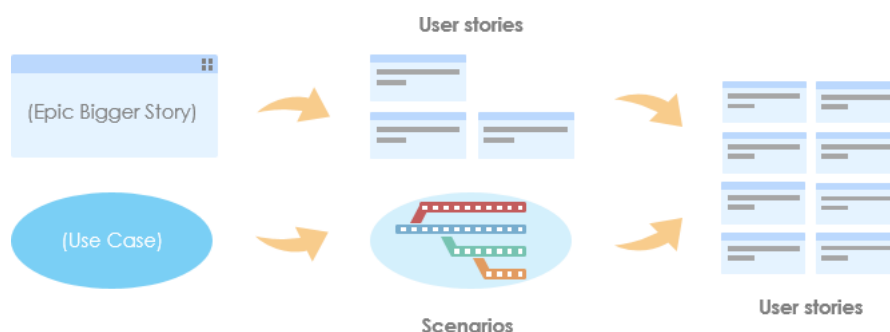
User Story is Now Mainstream

Anyone has the experience in software development would probably encounter the issues of communication between the end-user and the development team. That could be even more severe when members are working in different remote locations. User stories is a great way of opening discussion with clients for ensuring we really know what clients actually want. User stories created by the product owner capture "who", "what" and "why" of a requirement in a simple and concise way, which is typically written in natural language in non-technical format. Agile development has entered into the mainstream of development approach hand-in-hand with user stories for requirement discovery.

Practical Considerations

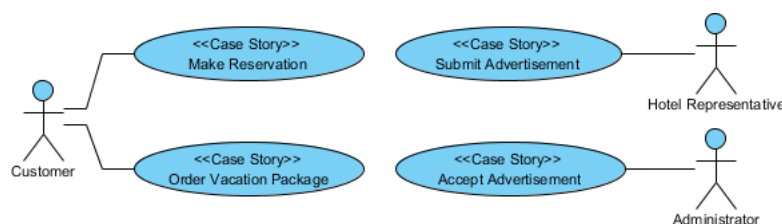
Typically, an Agile team with average of 10 members could end-up with hundreds of user stories in the working stream and some of them are inter-related resulting for the splitting from epics or the detail version of user stories from the previous Sprint. User stories are a transient artifact only resided in the Sprint and will be thrown away in the end of the development iteration. Agile team and Scrum member often found them easily gone unmanageable, hard to organize them a neat and orderly way, specifically team member would like to reference related user stories from the previous sprints.

While use case is meant to be more perpetual for the entire software development lifecycle and could be used as placeholder for accommodating the related user stories split within an epic scope. Moreover, the use case is meant to be continuous referenced by the development team for the subsequent development activities.



Use Case Diagram

A use case diagram is a kind of [Unified Modeling Language \(UML\)](#) diagram created for requirement elicitation defined by [Object Management Group \(OMG\)](#). Use case diagram provides a graphical overview of goals (modeled by use cases) users (represented by actors) want to achieve by using the system (represented by system boundary optionally). Use cases in a use case diagram can be organized and arranged according to their relevance, level of abstraction and impacts to users. They can be connected to show their dependency, inclusion and extension relationships. The main purpose of modeling use case with use case diagram is to establish a solid foundation of the system by identifying what the users want. Base on the result of analysis you can move forward to study how to fulfil those user needs.



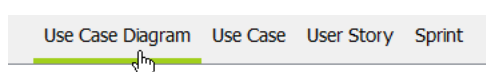
A use case diagram is mainly formed by actors, use cases and associations (connectors).

An actor is any person or external system that interacts with the system in achieving a user goal. There are two kinds of actors - primary and secondary. Primary actor is anyone or thing that interacts with the system to gain direct benefit. Secondary actor is anyone or thing that involve in achieving a use case yet, they do not gain direct benefit from the system. Very often, secondary actor is someone who assists the primary actor to achieve a use case.

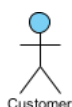
Drawing Use Case Diagram in Visual Paradigm

In this tutorial, we will make use of an online hotel reservation system as an example to demonstrate how to write effective use case with Visual Paradigm. Let's begin by drawing a use case diagram. We will carry on with writing effective use case with the resulting design.

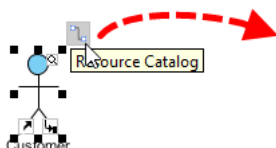
1. Open UeXceler in Visual Paradigm by selecting **UeXceler > UeXceler** from the application toolbar.
2. Open the **Use Case Diagram** page.



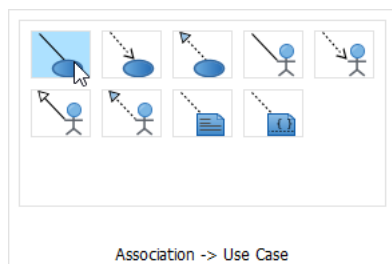
3. Select **Actor** in the diagram toolbar. Click on the diagram to create an actor and name it **Customer**.



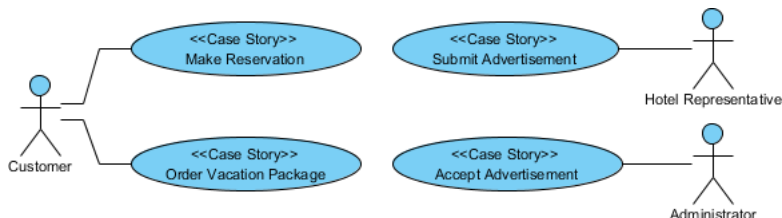
4. A customer can make a hotel reservation, which is a use case of the system. Let's create a use case from the *Customer* actor. Move the mouse pointer over the *Customer* actor. Press on the **Resource Catalog** icon at top right and drag it out.



5. Select **Association -> Use Case** in Resource Catalog.



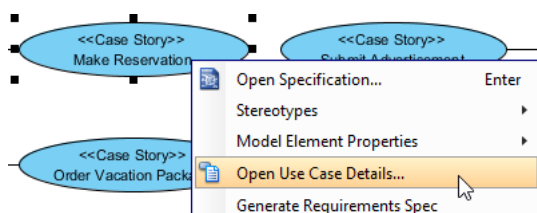
- Release the mouse button to create the use case. Name it *Make Reservation*. The association between actor and use case indicates that the actor will interact with the system to achieve the use case associated.
- Complete the design to make it look like this:



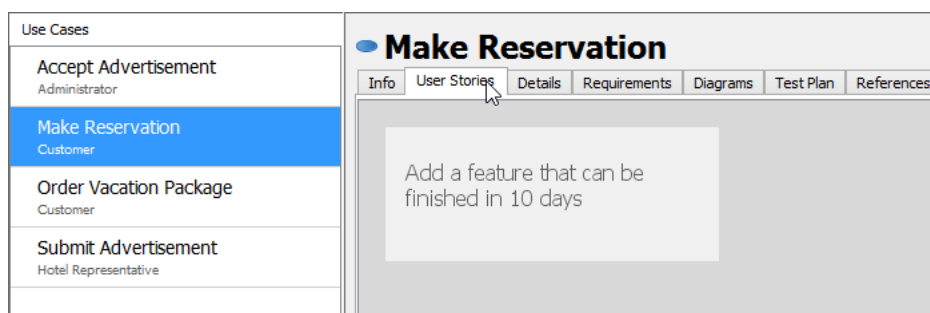
Write User Stories

While use case is the business goal of an IT system to be developed, user story represents a user problem or concern captured by the analyst and front-line stakeholders during the detailed discussion of a use case. No doubt that, all captured user stories aim to fulfill the business goal of the IT system.

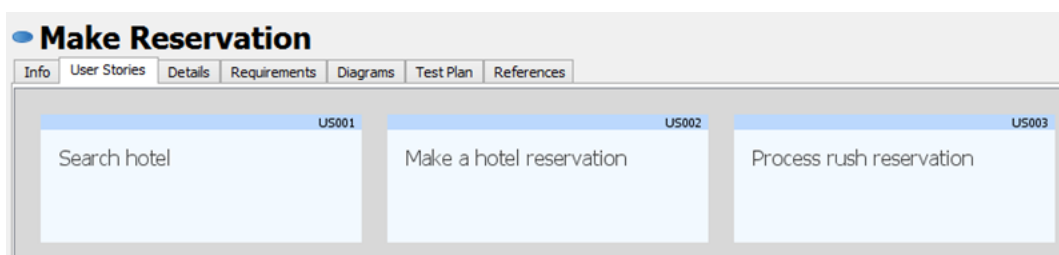
- Right click on *Make Reservation* and select **Open Use Case Details...** from the popup menu.



- Open the **User Stories** page.



3. Create user stories by double clicking the empty region inside the tab. Create three stories: *Search hotel*, *Make a hotel reservation* and *Process rush reservation*.



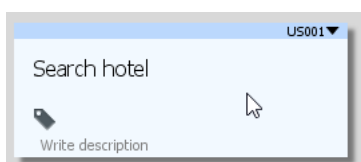
Capture User Story Scenario

A user story tells you what the end user wants to achieve by first identifying their problem. Once you have found out the problems, you can start looking for a solution. The user story scenario tool enables you to outline the interactions between actor and system in solving a problem as described under a user story. You can use this tool to find out the desired system behavior with user.

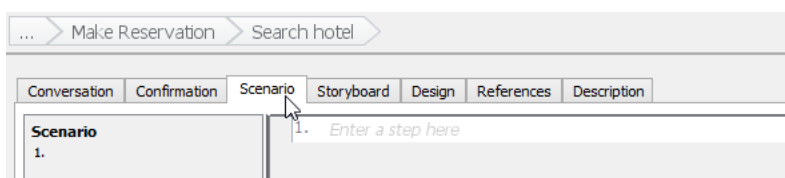
User story scenario constitutes a high level user-and-system conversation, which aims to find out the intents or actions of actor and how system react to those actor inputs. You should be concise when deciding what to include in the events flow. Do not include how system process user's input internally, or even implementation detail like an insertion of database record. This is wrong as user story, in fact use case analysis, is aimed to identify requirements from end user's point of view. Implementation details can, however, be modeled with [UML sequence diagram](#) in form of sub-diagram of user story.

Let's write the scenario of a user story.

1. Double click on the user story *Search hotel* to open it.



2. Open the **Scenario** tab. The scenario editor is formed by rows, known as steps. Each step represent an actor input or system response.



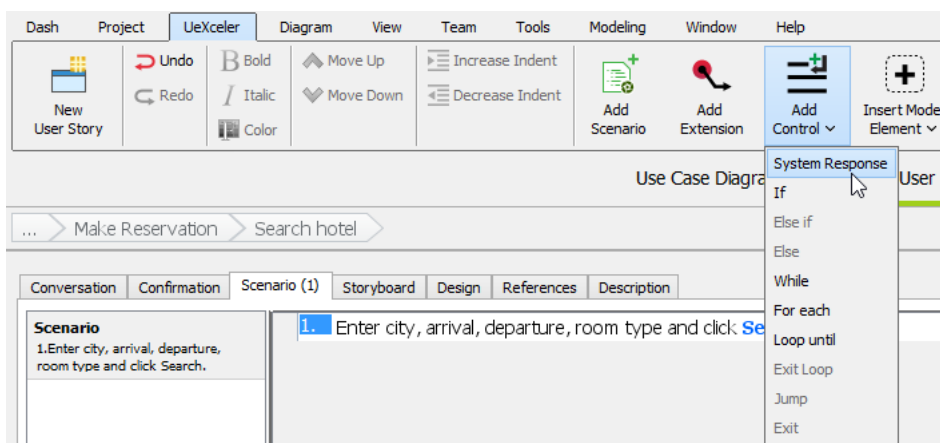
- Click on the first step and enter the first user input: *Enter city, arrival, departure, room type and click Search.*

1. Enter city, arrival, departure, room type and click Search.

- Use the format tools available under the **UeXceler** toolbar to set the word *Search* in blue and bold, for emphasis.

1. Enter city, arrival, departure, room type and click **Search**.

- Press **Enter** to complete this step. Step 2 will be created for you.
- Step 2 is about how the system react to user's input. You may start by writing "System...", but there is a better way to represent system response. Select **UeXceler > Add Control > System Response** from the toolbar to add a step of system response.



- You can now enter the content of step 2: *Display a list of hotels.*

1. Enter city, arrival, departure, room type and click **Search**.
2. **SYSTEM** Display a list of hotels.

- Add the following steps:

User input	System response
Click on a hotel's logo to read its detail	
	Display hotel details

1. Enter city, arrival, departure, room type and click **Search**.
2. **SYSTEM** Display a list of hotels.
3. Click on a hotel's logo to read its detail.
4. **SYSTEM** Display hotel details.

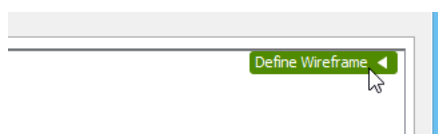
Something More - Creating Scenario-Based Wireframe

[Wireframe](#) is a sketch of user interface. It helps you represent the screen and screen flow of the system to be developed, early in requirements gathering. You can associate wireframes to steps in scenario. This section will show you how to make use of the [wireframe tool](#) to add a wireframe to a step.

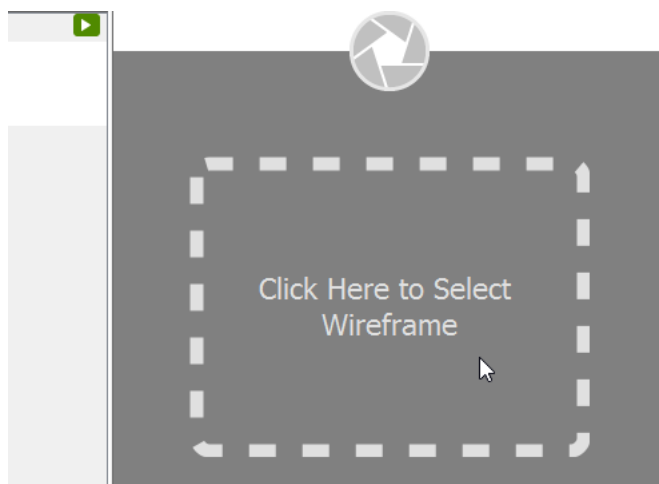
1. Click on the first step.

1. Enter city, arrival, departure, room type and click **Search**.
2. **SYSTEM** Display a list of hotels.
3. Click on a hotel's logo to read its detail.
4. **SYSTEM** Display hotel details.

2. Move your mouse pointer to the green triangle on the right hand side. Then click on **Define Wireframe**.



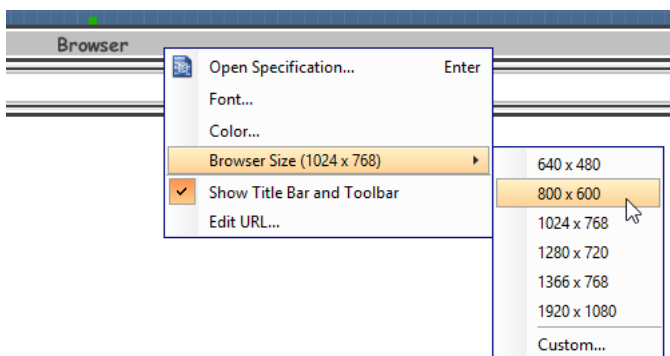
3. You see the gray pane appear on the right hand side? Click on it to select a kind of wireframe to create.



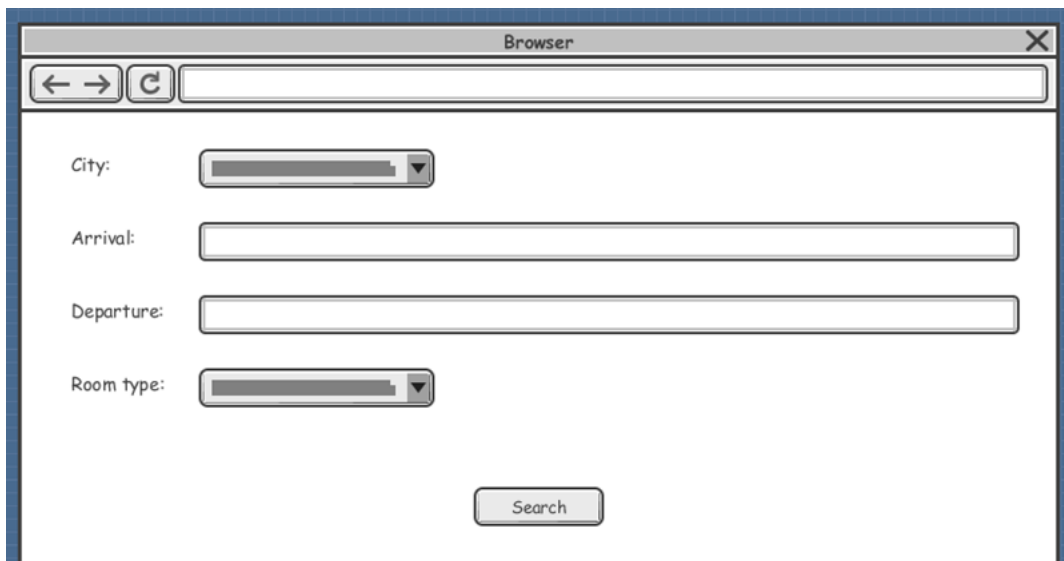
4. In the popup window, select **Website**.



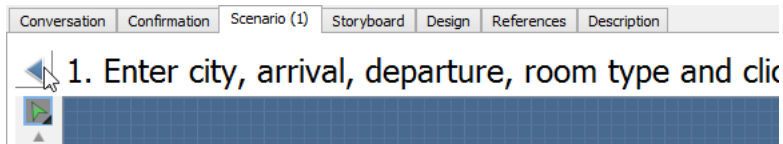
5. Click **New Website Wireframe**. A new wireframe appear, with an empty browser window in it. It's where you can prepare mockup for the web site.
6. Before we start adding different components into the Browser window, let's resize it to make it smaller. Click on the title of the Browser window.
7. Once clicked, the resize handlers appear around the Browser window for you to adjust the window size manually. Let's try a more direct method. Right click on the title of the Browser and select **Browser Size (1024 x 768) > 800 x 600** from the popup menu.



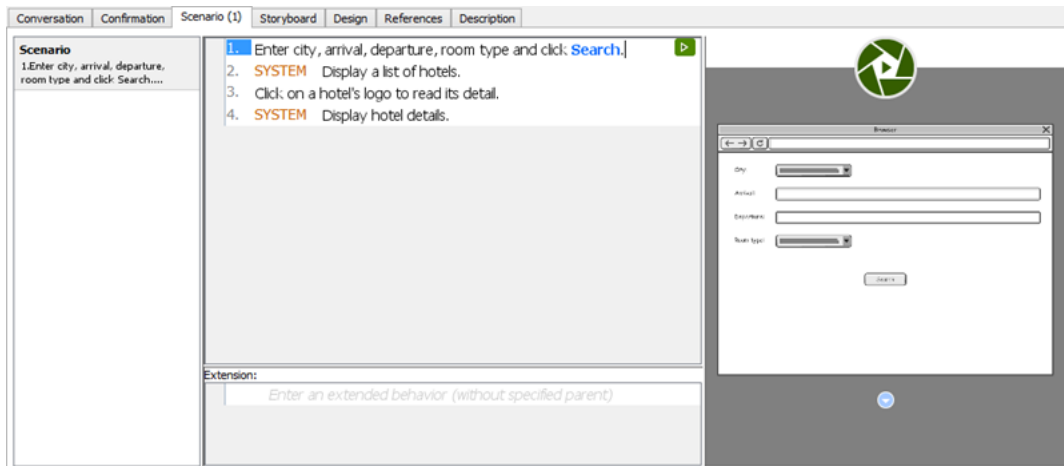
8. Make use of the wireframe tools listed in the diagram toolbar to create a wireframe like this:



9. Go back to the scenario editor by clicking on the triangle button next to the step title.



Finished, and you can see the thumbnail of your wireframe appear in the scenario editor.



Related Links

- [YouTube Video: Requirements Gathering with UeXceler](#)

- [YouTube Video: Elaborate Use Cases with Sub-diagram](#)
- [Wikipedia: Use case](#)
- [Wikipedia: Use-case analysis](#)
- [Wikipedia: Use-case model](#)
- [YouTube Video: How to Create Use Case Diagram from Business Process Diagram](#)



Visual Paradigm home page
(<https://www.visual-paradigm.com/>)

Visual Paradigm tutorials
(<https://www.visual-paradigm.com/tutorials/>)