


Use Case Narration

CSE 405N- Software
Analysis and Design

Md. Tanvir Al Amin Popel
S 2003 05 012

Introduction

Use case narration is a textual representation of the course of events encountered when an actor is interacting with the system. There can be several use cases associated with a system, each of which describes the system in a functional or behavioral point of view.

Use case modeling is an interface between the system developers and system users. The main requirement for a software project to be successful is building what actually is required to build. Clearly, the first step is requirements analysis, where analysts interact with end users and customers to clearly understand the system and both its functional and non-functional requirements. Use cases are a way to document these requirements systematically. Users can easily participate here and contribute to requirements analysis, because use case modeling is done in simple easy to understand human language.

There are two primary phases of use case modeling. One is use case diagram which just captures the requirements from a high level and illustrates what each actor can do with a subsystem.

Use case narratives are the next and most important phase, which describes each use case in detail as a path traversed through the system to meet a requirement. For the later technical steps use case narrations play the role of where to start. So, the more precise and complete an use case narration is, the more accurate will be the later designs like class diagrams, collaboration diagrams, sequence diagrams and state chart diagrams – which in turn rules how successful the project will be.

Purpose

Use case narrations help identify possible misunderstanding during very early stage.

As the course of events are written in detail as a form of communication between the actor any the system, use case narration often help to visualize the system in action, and is a meeting place of what the client want to get and what the developers think to build. This policy forces

the developers to focus on what the system must do, not how it is to be done, and avoids the trap of making assumptions about how the functionality will be accomplished.



Use case narration is a better replacement for older style specification sheets for a software system. Typical specification sheets generally contained several discrepancies because they didn't include end user-developer joint interaction and a small mistake in using the proper language resulted in a major fault in the developed system. Use case narrations document the requirements, with their perspectives and details of user-experience when using the system. So, use case narration is the way to user centric development.

A use case narration in formal style includes not only a typical success story of using the system, but also explicitly document several other point which help modern software processes to compare several implementation options come up with a better usable system even in a time boxed environment.

Parts of the Narration with example

Parts of narration vary with the style. There are several general conventions of writing narrations [AC1] . They are Brief Style, Casual and Fully dressed style.

Brief style is used generally in early requirements analysis phase, stories are written in short to get just for a quick sense of subject and scope [Lar1]. Casual style is written in a few paragraphs, which we can say a summary of the use case. The fully dressed style is most formal including all the supporting sections, both typical and alternate paths of traversal, preconditions, post conditions, constraints and so on. They are shown below. Here an example use case "Submit Order" is taken from our sessional project.

First five points are 1 Author, 2 Date, 3 Version, 4 Name. Date means the date of last modification. Version is the Current version of the use case. Use case name This should begin with a verb, the goal the use case is trying to achieve.

5 Type

An use case can be **Abstract** use case or **Extension** use case.

Or it can be **Business Requirement** use case, **System Analysis** or **System Design** use case.

Business oriented use cases take a high level view of the behavior and are free from technical details. **System use cases** are normally described at the sub process level and specify the data input and the expected data response.

Service Subsystem

Author (s): Group 5

Date: 01/02/2008

Version: 1.00

Use-case Name	Submit Order	Use-Case Type : Business Requirement
Use-Case ID	SML_07_01_A25	
Priority	High	
Source	SML_R1_05	

Fig 3.1 Initials of Use case narrative

The next 3 points are ID, Priority and Source. ID is a unique identifier. Priority helps to determine the order of implementation for each use case. Source can be a requirement, a specific document or a stakeholder, i.e. an entity that triggered the creation of the use case. These are illustrated in Fig 3.1

9 Primary Business Actor

Primary actor is the stakeholder who primarily benefits from the execution of the use case by receiving something of measurable or observable value.

Actor. Actors, in use case language, are agents (users or other systems) outside the system that interact with the system. Each actor defines a coherent set of roles users of the system can play (UML, 1999). Cockburn (1997) distinguishes between primary and secondary actors. A primary actor is one having a goal requiring the assistance of the system. A secondary actor is one from which the system needs assistance to satisfy its goal. On the other hand Whitten mainly distinguishes between Business actor, system actor, external server and receiver actors.

10 Other Participating Actors

Other participating actors include System actors, facilitating actors, server/receiver actors and secondary actors.



11 Interested Stakeholders

Those who are concerned about the use case

Primary Business Actor	Sales Officer
Other Participating Actors	Customer
Other Interested Stakeholders	Cashier – wants accurate fast entry and no payment errors Customer – wants fast service with minimal effort Company – wants to accurately record transactions and satisfy customer interests

Fig 3.2 Use case narrative Actors and Stakeholders

Description

A summary about what does this use case do.

Example **Description** : This use case describes the event of a sales officer submitting an order. The customer's detailed contact information and receivers information all are verified by the system. Once everything is verified and completed the order is accepted by the system and awaits for the accounts officer for payment. On completion the customer will receive an invoice

Precondition

Prcondition describes, what should be true before this user scenario can happen. Preconditions are assumed to be true inside the use case narrative. So they are not tested inside or violation of them are not kept as alternate path. Only noteworthy conditions which the author think the reader should be explicitly mentioned, are included

Trigger

The event that initiated the scenario

Precondition	The sales officer must be logged on
--------------	-------------------------------------

Trigger	The scenario starts when a customer enters the office and asks for a shipment.
---------	--



Fig 3.4 Use case precondition and trigger

Typical Course of Events

The main path to the goal. Typical course of events generally don't contain conditional events.

Alternate Courses

They are extensions to main path. Alternate courses can be noted in reference to the numbering done in Typical course of Events.

Example Typical and alternate course of events

Typical course of Events:

Step1: The customers arrives and informs the sales officer to submit order.

Step 2: The sales officer asks the customer the necessary details (destination, freight size, volume, type of goods) and enters the information into the system.

Step 3: The system checks if any required information is missing.

Step 4: The system then presents the payment charge to the officer

Step 5 : The officer asks the customer about the charge and payment details.

Step 6: When the customer agrees the officer submits the order to the system.

Step 7: The system then prints out invoice and and finishes submission of order.

Alternate course of events:

Alt-3. The officer did not enter all the necessary details.

The system again prompts for the missing information and submit again.

Alt-4. If the submission is not possible for shipping reasons.

The system informs about it to the officer and stops the submission of order.

Alt-7: The printer is out of paper

The system informs about it and the officer puts new paper.

Conclusion

Specifies when the use case successfully ends, i.e the primary actors receives something of measurable value.

Postcondition

Describes what should be true on successful ending of the use case, may be through typical course of events or alternate course.

Business Rules

The policies and procedures of the business the system should follow. These are the unique rules or methods of the business entity in discussion.

Conclusion : The use case completes when the officer successfully puts all the information in which case the system prints out a invoice.

Post condition: The order is marked as pending and is set for further processing by Shipping Officer, who makes the next deals.

Business rules :

1. There are 3 variations of Freight sizes. They are 18, 20 and 20H
2. The customer will have to take the invoice receipt to designated bank, and pay there.

Implementation constraints and specifications

And non-functional requirements and constraints are listed here. Example:

Implementation constraints and specifications:

1. Printing should be done on a laser printer, so that characters are easily readable, and high speed printing is possible.
2. The GUI should be user friendly and self descriptive.

Assumptions

Any assumptions that were made by the author.

Open Issues

Problems or matters yet to resolve. Example:

Open Issues: Need to resolve whether remote order placement will be supported



Variations

There are several variations in narrating the use case, the intention is same however. [AC2] enlists two basic styles of representation. One is plain text and another is in table format. [Wht1] shows a two column or conversational format – where the style of conversation between actor and system is explicit. In the samples section, samples are shown in different formats. [Hur1] discusses several dialects in Use Case Language.

We have found some variations in definitions also. Specifically definitions provided in [Wht1] are kind of different from other sources. Names of several points and their responsibilities can be different also. [Wht1] follows tabular format, but [Lar1] mainly follows the ways described in [AC1] and [AC2] with elaboration. There is also a variation about the placement of non functional requirements. Keeping these with use case narrative is not always suggested by practitioners [Lar1]. They suggest to move non-functional requirements in the supplementary specification, because it results in better content management, comprehensiveness and readability.

But during initial analysis, non-functional requirements should be kept with the use case narratives, because they help to take major architectural decisions.

Conclusion

Actually use case modeling is more of writing the textual descriptions or the user stories, because that part is important for taking decisions. Use case diagrams just give a overview of the possible scenarios in the system and their relationships. Narrations are the place where we concentrate on our business, what is happened in each scenario.

Here , in this document we have provided a brief sketch of use case narratives. The best way to use case narrations is via real world experience. As it is a practitioner's tool which help model the application better, the best format to use or best level of details to go are domain specific and depends a lot on context.



References

- [BJR1] Booch, G., I. Jacobson and J. Rumbaugh, The Unified Modeling Language User Guide. Addison-Wesley, 1999, pp.
- [Rub1] K. Rubin, and A. Goldberg, "Object Behavior Analysis", Communications of the ACM, Vol. 32, No. 2, Sept. 1992.
- [AC3] Cockburn, Alistair, "Structuring Use Cases with Goals", Journal of Object-Oriented Programming, Sep- Nov-Dec, 1997.
- [Jac1] Ivor Jacobson et al. Object-Oriented Software Engineering: A Use-Case Driven Approach, Addison-Wesley, 1992
- [Lar1] Craig Larman, Applying UML and Patterns, Third Edition
- [Ben1] Simon Benett, Steve McRobb, Ray Farmer, Object Oriented Systems analysis and design, Tata McGraw-Hill, New Delhi
- [AC1] Alistair Cockburn, Writing Effective Use Cases
- [AC2] Alistair Cockburn, Basic Use case template http://alistair.cockburn.us/index.php/Basic_use_case_template
- [Wht1] J.L Whitten, L.D. Bentley, K.C Dittman, Systems analysis and design methods, Tata McGraw-Hill, New Delhi
- [Wik1] Wikipedia Contributors, Use Case, http://en.wikipedia.org/wiki/Use_case
- [Hur1] Russell R. Hurlbut, A survey of approaches for describing and formalizing use cases <http://www.iit.edu/~rhurlbut/xpt-tr-97-03.html>
- [Hur2] Russel R. Hurlbut, The Three R's of Use Case Formalisms: Realization, Refinement, and Reification, <http://www.iit.edu/~rhurlbut/xpt-tr-97-06.html>
- [Brd1] Ruth Malan and Dana Bredemeyer, Functional Requirements and Use Cases, http://www.bredemeyer.com/pdf_files/functreq.pdf

Samples

Here 4 sample use case narrations are shown. To make the discussion clear, different formats are used.

Sample 1

Blog Subsystem

Author (s): Popel

Date: 30 April, 2008

Version: 1.0

USE CASE NAME:	Publish New Blog	USE CASE TYPE Business Requirements: <input checked="" type="checkbox"/> System Analysis : <input checked="" type="checkbox"/> System Design : <input type="checkbox"/>
USE CASE ID:	BLG-PNB	
PRIORITY:	High	
SOURCE:	Mr. Anis, Mr. Azom, Mr. Hasan	

PRIMARY BUSINESS ACTOR:	Member	
OTHER PARTICIPATING ACTORS:	Moderator, RSS Feed site	
OTHER INTERESTED STAKEHOLDERS:	Blog-Paper publishers	
DESCRIPTION:	This use case describes the process of publishing a new blog. The member can write the blog elsewhere and send the file, or he can write it in-place and then publish or he can do it via RSS feed. If documents contain pictures, a new scenario " <u>Insert Picture into Blog</u> " starts (extension). The written document is verified for abusive words according to member's permission levels, the blog is published directly or queued for moderation.	
PRE-CONDITION:	The member must be logged in The member should have permission to write blogs (must not banned)	
TRIGGER:	This user scenario starts when the member wants to write a new blog	
TYPICAL COURSE OF EVENTS:	Actor Action	System Response
	Step 1: Member wants to publish a new blogs	Step 2: System asks, about his preferred input method for the text and main language.
	Step 3: Member tells he wants to write it in-place using bangla language.	Step 4: System provides a default editor with cut-paste-bold-italic, insert picture and other formatting options, and sets the keyboard format to bangla.
	Step 5: Member writes the text, and at some point wants to insert a picture	Step 6: Extension use case scenario " <u>Insert Picture into Blog</u> " starts
	Step 7: Member applies some formatting like text resize, italic, bold etc.	Step 8: System applies the formatting to document.
	Step 9: Member completes and wants to publish it	Step 10: To be sure that member is a human, not a computer program, system authenticates member. Extension point : <u>Authenticate by Challenge Image</u>

		Step 11: Member successfully authenticated himself. and Member has instant-publish permission. So the blog is published. A confirmation is generated and shown to member.
ALTERNATE COURSES:	<p>Alt – 3a : Member wants to publish by file. New user scenario “<u>Upload Blog File</u>” starts.</p> <p>Alt – 3b: Member wants to publish via RSS Feed. New user scenario “<u>Get RSS Feed</u>” starts.</p> <p>Alt – 3c : Member wants English as language System keeps the keyboard input English.</p> <p>Alt -9a : Member don’t want to publish blog right now. He wants to save and publish later. System saves the blog in file.</p> <p>Alt – 11a: Member failed to authenticate <u>Challenge Image</u>. He is given some number of chances to reauthenticate. If he ultimately fails, the account is blocked in the confusion of script-based-access.</p> <p>Alt 11-b : Member doesn’t have instant-publish permission. He has “loose moderation” option. The blog is verified of some abusive words. If they are not found it is published.</p> <p>Alt 11-c : Member has strict moderation account. The blog is kept pending for human level moderation. Publish is delayed until moderator confirms.</p>	
CONCLUSION:	The use case concludes, when the member receives the confirmation about blog published.	
POST-CONDITION:	A news about this blog is shown in “Latest Blogs” section at home page, if the blog is published successfully.	
BUSINESS RULES	<ul style="list-style-type: none"> • The preferred languages are bangla (Unicode) and English. • Preferred input methods are In-Place, File or RSS Feed • For file inputs, doc, rtf, html files are supported • For bangla text, Avro, National and Phonetic keyboard are supported. 	
IMPLEMENTATION CONSTRAINTS AND SPECIFICATIONS	<p>The moderators and members, all have Web based thin client.</p> <p>The database must support Unicode, because here bangla can be used.</p>	
ASSUMPTIONS:	<ul style="list-style-type: none"> • Member has bangla fonts. • Member has modern browser installed. 	

	<ul style="list-style-type: none"> • Moderators will be notified by email, if the member has strict moderation option.
OPEN ISSUES:	<ul style="list-style-type: none"> • Whether other languages will be supported is an open issue. • How many pictures in a blog entry is supported. • Loose moderation by just scanning for abusive words is not a good idea. An amendment to this is in process.

Sample 2 : This is a use case from our SADD Project

Use Case: 10 Assign Worker

CHARACTERISTIC INFORMATION

Goal in Context: Workers are assigned to empty slots to work there.

Scope: Shipping Subsystem

Level: Summary

Preconditions: We know the workers assignment done, and also know the slots empty.

Success End Condition: Worker has assigned slot.

Failed End Condition: Worker already is assigned or the slot has other workers to work with

Primary Actor: Shipping Personnel.

Trigger: Shipping Personnel wants to process the slot by assigning worker.

MAIN SUCCESS SCENARIO

1. Shipping personnel selects a slot currently being processed.
2. Shipping personnel checks the workers already assigned there and volume of freights allocated to that slot
3. If more worker is needed, shipping personnel finds unassigned worker for that date
4. Shipping personnel selects one worker for that slot from list of unassigned workers.
5. Worker is notified via manual call.

EXTENSIONS

- 3a. No more worker is needed for that slot : No worker is assigned for that slot
- 3b. No unassigned worker is found : A request for new worker intake is sent

SUB-VARIATIONS

- 5 Worker may be notified by
Directly calling them
By supervisor
Via phone

RELATED INFORMATION

Priority: Top

Performance Target: Instant worker assignment

Frequency: Daily basis

Superordinate Use Case: Create Slot (use case 8) , Submit Order (use case 2) , Process Order 3)

Subordinate Use Cases: Generate Worker history (use case 22), Generate worker wages (18)

Channel to primary actor: Interactive, email, phone

Secondary Actors: Worker

Channel to Secondary Actors: Interactive, Phone, Via supervisor

OPEN ISSUES

What if worker is not accessible after assignment ?

SCHEDULE

Due Date: 5 April, 2008

Sample 3 : This is an extension use case called in Sample 1

USE CASE #	20 Authenticate by Challenge Image
Goal in Context	When a user is going to do some major change, we need to verify it is done by a human, not a malicious script
Scope & Level	System, Subfunction
Preconditions	User is already logged in via username-password method.
Success End Condition	User is authenticated as human
Failed End	We can't agree his reading of the picture with our saved value. User can't

Condition	authenticate.	
Primary, Secondary Actors	User Dynamic-Challenge-Image Generator	
Trigger	When user does something that initiates a major change (new user, new blog, new post, adding new friend etc)	
DESCRIPTION	Step	Action
	1	A challenge image is generated by challenge-image generator, and it is shown to user.
	2	User inputs the text or number in the image.
	3	System checks whether it matches with the value saved for that challenge-image
	4	User input text matches, and the user is authenticated as human
EXTENSIONS	Step	Branching Action
	4a	User input text doesn't match : User can't authenticate
SUB- VARIATIONS		Branching Action
	1	The challenge image generated may contain noise added pictorial representation of "0-9" "A-Z" or "a-z".
	2	The challenge image will be 80X80 or 120X120 or 320X240 size.

RELATED INFORMATION	
Priority:	Medium
Performance	10 second to generate image, 2 minute to submit authentication.
Frequency	300/day
Channels to	Email

actors	
OPEN ISSUES	What if making the challenge-image generator is complex ? Shall we rely on web basee challenge-image generators ?
Due Date	10 May, 2008
Superordinates	Publish Blog, Create New User, Publish Story, Upload to Gallery
Subordinates	<optional, depending on tools, links to sub.use cases>

Sample 4 : This is an extension use case called in Sample 1

Blog Subsystem

Author (s): popel

Date: 1 May, 2008

Version: 2.00

USE CASE NAME:	Insert Picture into blog	USE CASE TYPE Business Requirements: <input checked="" type="checkbox"/> System Analysis: <input checked="" type="checkbox"/> System Design: <input type="checkbox"/>
USE CASE ID:	BLG-IPB	
PRIORITY:	Medium	
SOURCE:	Mr. Azom, Mr. Khosru, Mr. Jubaer	
PRIMARY BUSINESS ACTOR	Member	
OTHER PARTICIPATING ACTORS:	Moderator, Photo Gallery (Other subsystem)	
DESCRIPTION:	This use case describes the scenario when an user wants to attach a photo/picture with a blog entry. The photo will be uploaded to our gallery, the link will be used to show the picture.	
PRE-CONDITION:	The user already have access our external site The user has permission to attach photo with blogs	
TRIGGER:	The scenario starts when, writing a new blog in-place user wants to attach a picture with it	
TYPICAL COURSE		
	Actor Action	System Response

OF EVENTS:	Step 1: Member wants to attach a photo at preferred place of the document.	Step 2: System asks for the preferred input method.
	Step 3: Member selects directly upload from pc	Step 4 : System asks for the path of the file in computer.
	Step 5: Member selects the photo file and gives permission to access file system	Step 6: System uploads the photo and checks whether it is a valid photo file, whether it has size and dimension under limit.
		Step 7: Depending on the users' permission level, the photo is selected for moderation or directly inserted.
		Step 8: The link of the picture is placed at the appropriate position of blog text.
ALTERNATE COURSES:	Alt 3a : Member wants to upload from another site. System asks for the link of that site, and takes photo from that link	
	Alt 3b: Link provided by member doesn't work .Member is notified.	
	Alt 6a: Photo exceeds size limit, or is not a valid picture. >> Member is notified.	
CONCLUSION:	The use case ends when the photo is saved and link is placed on text.	
POST-CONDITION:	If the member does have strict moderation options, moderators are notified to check that photo.	
BUSINESS RULES	<ul style="list-style-type: none"> • JPG or GIF images are supported only • Max image size 256 KB • Max dimension 800X600 • Image shouldn't contain offensive materials • Image shouldn't be a copyrighted image. 	
CONSTRAINTS SPEC.	The moderators and members, all get Web based thin client	
ASSUMPTIONS:	<ul style="list-style-type: none"> • Moderators will be notified by email, if the member has strict moderation option. 	
OPEN ISSUES:	What if RTF file inputs contain, embedded copyrighted images ? What if the link provided by the user contains copyrighted materials ?	