Use Case to Object-Oriented Code Base

Presenters:  Marilou Landes (ACES)
Melissa Rubik (OBFS)
Leslie Sherman (OBFS)
Overview

- Introduction to applications
- Development methodologies
- What is a use case?
- How to convert a use case to a model
- How to convert a model to code
- Using use cases for testing
- Questions

Use Case to Object-Oriented Code Base
IPHE Procurement Bulletin

- Online resource for posting and finding upcoming bids, requests for proposal, requests for information, solicitation documents, and awards for IPHEC universities.
- Public site for vendors
- Internal site for purchasing staff
- [http://www.procure.stateuniv.state.il.us](http://www.procure.stateuniv.state.il.us)
Use Case to Object-Oriented Code Base
IPHE Procurement Bulletin: Internal Site

---

**Illinois Public Higher Education Procurement Bulletin**

### To Do

- **Create New**
- **Award/Change/Renew**
- **Resources**

#### Quick Search:

```
Search
Advanced Search
```

**Notice Status** | **Document Management** | **Contact Us** | **Log Off**

During peak times it may take several minutes for new items to appear on your To Do List.

### You currently have no notices to do.

#### GSU Needs Award

<table>
<thead>
<tr>
<th>Procure #</th>
<th>Description</th>
<th>Opened</th>
</tr>
</thead>
<tbody>
<tr>
<td>QBS1134</td>
<td>architectural servicing</td>
<td>Aug 10, 2010</td>
</tr>
<tr>
<td>D1145TMS</td>
<td>Implementation of Ex...</td>
<td>Apr 18, 2011</td>
</tr>
</tbody>
</table>

#### Needs Renewal?

<table>
<thead>
<tr>
<th>Procure #</th>
<th>Description</th>
<th>Ended</th>
</tr>
</thead>
<tbody>
<tr>
<td>D8143TMS</td>
<td>Project Management Consulting</td>
<td>Jun 30, 2011</td>
</tr>
<tr>
<td>D8169TMS</td>
<td>Service Contract for...</td>
<td>Jun 30, 2011</td>
</tr>
<tr>
<td>D8156JCG</td>
<td>Copier upgrade and service</td>
<td>Jun 30, 2011</td>
</tr>
</tbody>
</table>

The public web site is at [http://www.procurate.stateuniv.state.il.us](http://www.procurate.stateuniv.state.il.us)

---

Use Case to Object-Oriented Code Base
eText

- eText is a vehicle for presenting a textbook to instructors/students for use inside/outside the classroom
- Each person can "own" one or more books
- Instructors can make assignments; students can complete assignments
- Instructors/students can make private or public notes; public notes can be used as a discussion forum
- URL: https://etext.illinois.edu
Use Case to Object-Oriented Code Base
Use Case to Object-Oriented Code Base
Primary Sources: Documents and Interviews

In research writing, there are two kinds of sources of information: primary and secondary. Primary sources are records—or witnesses—to history, either first-hand accounts of events, or historical physical artifacts themselves. They include the objects that surround you every day: photographs, memos, diaries, receipts, and clothing. Your toothbrush, cell phone, papers you have written, including those for this class, might all be primary sources, depending on the research question brought to them.

Secondary sources will likely also be very familiar to you. They include all textbooks, books of history, articles written by academics—in other words, most things that you have read in school up until this point. Secondary sources are helpful because the author has already digested a number of primary sources for you in order to present a coherent slice of history or to make a claim. However, although secondary sources might seem the easiest place to look for inspiration in research, in fact, primary sources can often be the most inspirational.

Reading and Writing Exercises

As you read the primary sources research narrative written by former Rhetoric 105 students Christopher DeLetto and Matthew Rasinski, take note of the many ways a variety of primary sources inspire new questions and potential avenues of research.

How many primary sources are cited on the second and third pages of the essay? How do these sources connect to the argument? What is your reaction to the writers’ six-point history of the university’s attempts to eliminate racial segregation in campus housing? How else could the writers have integrated this data with their argument? Do you think DeLetto’s and Rasinski’s conclusions are reasonable? What other questions or
Development Process Flows

IPHE Procurement Bulletin

Use Case to Object-Oriented Code Base
Development Methodology

- Bulletin – more structured
  - Have all artifacts
  - Team communications – SharePoint
  - Roles were done by different team members

- eText – less structured
  - Verbal discussions and few artifacts
  - Team communications – Basecamp
  - Roles shared by one or more team members
What is a Use Case?

- Definition:

  “A use case in software engineering and systems engineering is a description of steps or actions between a user (or "actor") and a software system which lead the user towards something useful.”

  -- Wikipedia

---

Use Case to Object-Oriented Code Base
Why Do We Need Use Cases?

- Communication tool with Users to document user behaviors
  - Contract with Users of the system’s behaviors
  - Use to assist with change control (avoid scope creep!)
- Assists with developing Model
- Assists with developing UI design
- Basis for Test Plans
Use Case 1: Create Reminder

1. Buyer starts an award or renewal on a notice.
2. System displays preview of award or renewal.
3. Buyer updates reminder date and buyer name and submits award or renewal.
4. System saves reminder information.
5. Use case ends.

Use Case 2: View Reminder List

1. Buyer or Site Contact chooses to view their reminder list.
2. System displays list.
3. Buyer or Site Contact selects a notice from the reminder list.
4. System returns preview of the current version of the notice.
5. Buyer or Site Contact chooses to review the award.
6. System starts renewal process.
7. Use case ends.

Use Case to Object-Oriented Code Base
Use Case 1:
1. When an instructor makes an assignment,
   - it can be visible immediately, or in the future
   - it is visible to students in the instructor's section and to the book's author(s)
   - it can be modified or deleted if it's not yet visible
2. When a student responds to an assignment, only the instructor(s) of the section and the book's author(s) can see the response.

Use Case 2:
1. When a student creates a public note, it is visible immediately to the book's author(s) and the student's instructor(s);
What is a Model?

- **Definition:**
  
  “a class diagram in the [Unified Modeling Language (UML)](https://en.wikipedia.org/wiki/Unified_Modeling_Language) is a type of static structure diagram that describes the structure of a system by showing the system's [classes](https://en.wikipedia.org/wiki/Class_%28computer_science%29), their attributes, operations (or) methods and the relationships between the classes.”

  -- Wikipedia

- A tool to understand the system under development
- It is a living document.
Convert Use Cases to a Model

- Identify nouns and verbs
- Nouns
  - Classes (People, Places, Things)
  - Properties
    - Do obvious ones
    - Add others later
  - Relationships between classes
- Verbs
  - Methods
Convert Bulletin Use Cases to a Model

Use Case 1: Create Reminder
1. Buyer **starts** an award or **renewal** on a **notice**
2. System displays **preview** of award or renewal
3. Buyer updates **reminder date** and **buyer name** and submits award or renewal.
4. System **saves** reminder information.
5. Use case ends.

Use Case 2: View Reminder List
1. Buyer or Site Contact **chooses to view** their reminder list.
2. System **displays list**.
3. Buyer or Site Contact **selects a notice** from the reminder list.
4. System **returns preview** of the current version of the notice.
5. Buyer or Site Contact **chooses to review** the **award**
6. System **starts renewal** process.
7. Use case ends.

Use Case to Object-Oriented Code Base
Convert eText Use Cases to a Model

Use Case 1:
1. When an instructor makes an assignment,
   - it can be visible immediately, or in the future
   - it is visible to students in the instructor's section and to the book's author(s)
   - it can be modified or deleted if it's not yet visible
2. When a student responds to an assignment, only the instructor(s) of the section and the book's author(s) can see the response.

Use Case 2:
1. When a student creates a public note, it is visible immediately to the book's author(s) and the student's instructor(s);
public class Reminder extends Object {

  **
  * a user object representing the person to be contacted */
  private PrivateUser contact;
  /**
   * the reminder date */
  private Date reminderDate;

  public Reminder(){
  }

  /**
   * This method initializes and returns the Reminder object.
   * @param reminderDate (optional) the reminder date
   * @param contact (optional) a user object representing the person to be contacted
   */
  public Reminder init(Date reminderDate, PrivateUser contact){
    return null;
  }

  /**
   * The method makes sure the information satisfies business rules.
   * @param errorIniPath the path to the Errors.ini file
   */
  public string[] validateData(string errorIniPath){
    return null;
  }
}
Convert a Model to Code - ColdFusion

<!--- Reminder.cfc
Purpose: I am a reminder. I have a date and contact information.
Last Modified: $Date$ - $Rev$ by $Author$ - $URL$
extends="webdata.common.object"
--->

<cfcomponent name="Reminder" extends="webdata.common.object" hint="I am a reminder. I have a date and contact information.">
<!--- contact - a user object representing the person to be contacted --->
<cfset variables.instance.contact=""
<!--- reminderDate - the reminder date --->
<cfset variables.instance.reminderDate=""

<cffunction name="init" access="public" returntype="Reminder" output="false" hint="This method initializes and returns the Reminder object.">
<cfargument name="reminderDate" type="Date" required="false" default="" hint="(optional) the reminder date">
<cfargument name="contact" type="PrivateUser" required="false" default="" hint="(optional) a user object representing the person to be contacted">
<cfreturn this />
</cffunction>

<cffunction name="validateData" access="public" returntype="string[]" output="false" hint="The method makes sure the information satisfies business rules.">
<cfargument name="errorIniPath" type="string" required="false" default="" hint="the path to the Errors.ini file">
<cfreturn [fill in your string[] here] />
</cffunction>

</cfcomponent>
Using Use Cases for Testing

- Basis for testing flows
  - Unit testing
  - Integration testing
  - User acceptance testing

- When application passes the Use Case flows you have ensured that the behavior matches customer expectations
Questions?
References

- http://www.procure.stateuniv.state.il.us
- https://etext.illinois.edu