Com S/Geron 415X Gerontechnology in Smart Home Environments

Use Case and Workflow

Dr. Hen-I Yang
Computer Science Department, ISU

March 24, 2011
Announcement

- March 29 lecture will be held in Design 234
  - Bring your preliminary work for Homework 4 (identity design and branding) to class
  - Bring your notepad to take notes on design students’ presentation
- Remember to continue weekly meeting and post minutes
- Suggested progress of the week:
  - Identify all the use cases for your system
  - Identify the use cases you plan to implement in your prototype
  - Devise the workflow(s) for your system
- Reading: Chapter 31 in the Sommerville book (the book in Reserve at SHL)
Midterm Report

- Due: April 5, 2011 by midnight via WebCT
- Single page document
  - Response to comments
  - Changes made to the project since proposal
  - Summary of progress since proposal
  - Links to a collection of web-based documents
- Expected Materials to be posted on the PBWorks project website
  - Updated Gantt Chart
  - Use Cases and Workflow
  - Identified Services (Interfaces and Metadata)
- A neatly designed, comprehensive PBWorks project website
- The primary focus is to evaluate if you have made reasonable progress
Previously in 415x …

- Introduction to Project Management
- Brief introduction to software engineering
- Storyboarding
- Actors
- User Scenarios

- How many user scenarios have you identified?
- Tips: Keep a record of all the outcomes from the discussion both in-class or outside of the classroom, and include them in your midterm/final reports.
Objectives

- Learn about how to define use cases for C-requirement
- Learn about how to use workflow to describe (business) process
- Learn about how workflow can be used to help design the software system
Use Case
Use Cases

- A use case describes
  - sequences of actions
  - initiated by an actor that
  - a system performs that
  - yield an observable result of value

- How are use cases different from user scenarios?
  - Are generalized (e.g. use of actor roles instead of the name of a specific user, use of a category of options instead of a particular choice)
  - Provide additional information (e.g. pre/post conditions)
  - Often merge alternatives and exceptions of related scenarios into one single use case
Use Cases

- Elements of use cases
  - Actors (users, other systems or applications, devices)
  - Use Case
    - Name
    - Brief Description
    - Actor
    - Flow of Events
    - Pre-conditions
    - Post-conditions
    - System/subsystem
    - Other stakeholders and special requirements
Benefits of Use Cases

- Relatively easy to write and read
- Force developers to think through the design of a system from the perspective of a user
- Engage the users in the requirements process, helping them understand the system and giving them a way to communicate and document their needs
- Give context for the requirements (why and how)
- Provide an ordering mechanism for requirements, to tell what has to happen before the next thing happens
- Can carry over directly into the testing process
- Serve as inputs to the user documentation
- Provide a convenient step-by-step format
A Sample Use Case for Retriever

Name: Lookup Lost Item

Brief Description:

User wants to find one of the pre-defined item lost in the room with the help from the system

- **Actors:** User

- **Main Flow:**
  1. User selects from the lost item from a list of the pre-defined items by pressing a button on the GUI
  2. The system shows instructions on how to upload a photo on screen
  3. User takes a picture of the room where the lost item may be found
  4. User selects the photo of the room using the file browser window (Windows Explorer)
  5. User presses the “upload” button on the GUI
  6. The system starts analyzing the picture to search for the lost item while displaying the spinning wheel
  7. The system displays the uploaded photo, with the red square highlight around where the wallet is in the photo
A Sample Use Case of Retriever (con’t)

- **Alternate Flow 1:**
  1. User selects from the lost item from a list of the pre-defined items by pressing a button on the GUI
  2. The system shows instructions on how to upload a photo on screen
  3. User takes a picture of the room where the lost item may be found
  4. User selects the photo of the room using the file browser window (Windows Explorer)
  5. User presses the “upload” button on the GUI
  6. The system starts analyzing the picture to search for the lost item while displaying the spinning wheel
  7. The system displays the error message “the item cannot be found at this location” on screen
A Sample Use Case of Retriever (con’t)

**Alternate Flow 2:**
1. User selects from the lost item from a list of the pre-defined items by pressing a button on the GUI
2. The system displays a list of surveillance cameras available in the room
3. User selects from the list of surveillance cameras
4. The system uses the selected camera to take a picture of the room
5. The system starts analyzing the picture to search for the lost item while displaying the spinning wheel
6. The system displays the error message “the item cannot be found at this location” on screen

**Preconditions:**
- The item lost has been programmed/trained as an object recognizable by the system
- The user has the capability to supply photos in JPEG or GIF formats
Step-by-step Guide – Use Case

1. **Identify and describe the actors**
   - Who uses the system? Who gets/provides information? Who supports and maintains the system?

2. **Identify the use cases and write a brief description**
   - What will the actor use the system for? Will the actor create, store, change, remove or read data in the system? Will the actor need to inform the system or be informed about external events?

3. **Identify the actor and use-case relationships**
   - Examine if additional actors are associated with each use case

4. **Outline the individual use cases**
   - Determine the “flow”
   - Sunny day scenario and alternate flows

5. **Refine the use cases**
   - Determine all the alternate flows
   - Determine pre- and post- conditions
Use Case Model: Medication Information Support Systems (MISS)

- **doctor**: Enter prescription, Check for Conflict, Confirm Prescription data
- **pharmacist**: Fill Prescription
- **patient**: Check-in Medication, Take Medication, Receive Warning/Update, Dispense Medication
- **Smart home**: Receive Warning/Update, Dispense Medication
In-Class Activity: Define Use Case

- Review the user scenarios that you identified
- Group related scenarios based on particular use cases that an actor will use your system for
- Merge these scenarios into a use case, remember to include name, brief description, actor(s), main and alternative flow(s), and pre-condition(s)
- Draw use case model to see how actors are connected in various use cases
- *You should identify as many use cases as possible, you can then identify which use cases you will implement in your prototype in the midterm report*
Workflow
Workflow

- A workflow diagram is a graphic representation of all the major steps of a process. It can help you:
  - Understand the complete process.
  - Identify the critical stages of a process.
  - Locate problem areas.
  - Show relationships between different steps in a process.
Example (Workflow)
What is workflow?

- Used for modeling the behaviors of the system
- Creating Workflows from Use Cases
  - Start from one use case – one workflow
  - Explore the critical internal point (activities and events) of the system/service
  - Identify the interaction pattern between the actor and various parts of the system
  - Examine the interfaces needed (GUI/ Service interface/ or others)
Workflow (BPMN style)

- Activities
- Events
- Gateways (choices)
- Sequence of activities
- Message exchanges

Hotels.
GetRequirementsf

Hotels.
CheckAvailability

Hotels.
ConfirmReservation

Hotels.
NoAvailability

Hotels.
ReserveRooms

Customer
Interacting Workflows

SetupComputation

- Request Processor
- Setup Job Parameters
- Download Data
- Start Computation
- Restart
- Store results
- Report Completion

VectorProService

- Check Availability
- Allocate Resources
- Initialize
- Compute
- Return Results

Report Completion

Completion

Setup Job

Parameters

Start Computation

Download Data

Restart
Example Interaction Workflows (Retriever)

Retriever

Target Item Selection
Display Photo
Upload Instruction
Display Photo
Upload Interface
Analyze Photo

Inquire whether to try a different photo
Display Search Result

User

Select Item To Be Found
Take photo of the room
Upload photo
Evaluate the result

Pickup the missing Item
Workflow Refinement

- Check interfaces and communication channels
- Can be an iterative refine process
  - Decomposition:
    Can start with the system as a whole, then breakdown into services/components
  - Merger:
    Examine all workflows created, merge them if there are causal relations between activities in different workflows
Example: Decompose Workflow (Retriever)

OpenCV

Retriever

User

- Target Item Selection
- Display Photo
- Display Photo Upload Instruction
- Locate Item on the Photo
- Inquire whether to try a different photo
- Display Search Result
- Select Item To Be Found
- Take photo of the room
- Upload photo
- Evaluate the result
- Pickup the missing Item
In-Class Activity: Design Workflow

- Pick one of the use cases you defined, identify critical activities and events in the main flow for both the user and the system
- Identify the dependency and the sequence of the activities and events
- Draw the workflow based on the information you just identified
- Add the messages/calls into the workflow
- Look at the alternative flow(s) and modify your workflow to include new sequence of activities
- Decompose your system workflow into independent services
Next Time in 415x

- Learn what Sequence Diagram is
- Learn how to realize the user requirement and how to design and implement services
  - System implementation by orchestrating existing and new services
  - Design and implement service following the guidelines of service engineering