# Principles of GUI Design and Programming

Task Descriptions

# Contents

- Scenarios
- Use cases

# Task Descriptions

- Now that we have descriptions of prototypical users, we need descriptions of how they will use the system
- By knowing the users and how they will use the system, we can design an effective interface for them
- We will look at
  - Scenarios
  - Use cases

# Scenarios

- A scenario is an
  - ‘informal narrative description’ (Carroll, 2000).
- It describes
  - human activities or tasks in a story that allows exploration and discussion of
    - contexts, needs, and requirements.
- It does not necessarily describe the use of software or other technological support to achieve a task.

# Understandable by all

- Since scenarios are informal descriptions of tasks
  - Stakeholders participate in their development
  - They are written in the language and terminology used by the stakeholders
  - This makes them understandable by everyone
- Scenarios avoid the use of technical terms that would not be familiar to the stakeholders
- Scenario writing is often the first step in gathering requirements

# Scenario Example

- Performing data entry for an admissions office

Well, this is where we process admissions forms. We receive about 50 a day during the peak application period. Brian here checks the applications to see that they are complete, that is, that all the information and supporting evidence has been included. You see, we require evidence of relevant school exam results or work experience before we can process the application. Depending on the result of this initial inspection, the applications are sent on to . . .

# Scenario writing

- Each scenario describes one task the stakeholder performs
- You will usually develop many scenarios when describing the tasks for an entire system
- Each scenario describes one common use for the system

# Scenario writing

- Scenario writing is really just story telling
- You are telling someone the story of how you perform a task
- The level of detail varies depending on where the scenario is being used in the development process
- During requirements it is a good idea for scenarios to emphasize
  - the context,
  - the usability and user experience goals,
  - and the tasks the user is performing.

# Scenario writing

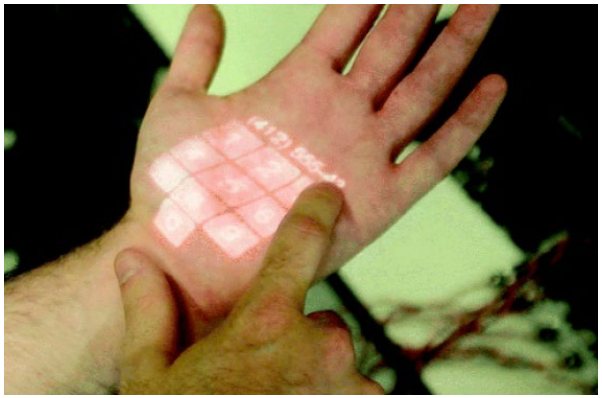
- Scenarios are often developed during
  - Workshops,
  - Interview,
  - Brainstorming sessions
- They can capture
  - Existing uses of the system
  - How people envision the system being used in the future

# Envisioned scenarios

- These describe a task that has never been done
- It is a task that will be performed by a system that have yet to be developed
- It might possibly describe the use of new technology being employed for the first time

# Envisioning skin input device

- Bramat has just finished his daily 4 mile run. He likes listening to music while he exercises, and has been playing his favorite pieces. This new skinput technology is great as he can focus on the running while scrolling through the available tracks, skipping through them with a simple tap of his fingers. He comes in exhausted and flops down on his favorite seat. With a flick of his fingers he turns off his music player and opens the palm of his hand to reveal the television remote control panel, graphically projected on his skin. He taps on a button to choose the station for the program he wants, adjusts the volume with a few more taps, and sits back to watch. Feeling hungry, he walks to his kitchen, opens his palm once again and sees a list of recipes possible given the items in his fridge. With another hand gesture, his palm turns into a telephone keypad, from where he can invite a friend over for dinner.



# Use cases

- In the beginning, requirements were developed by
  - A group of developers or designers sitting in a room writing them down
  - They often had little or no contact with the users of the system
  - They required the features they thought the users wanted, not what the users actually wanted

# Use cases

- Ivar Jacobson was working at Nokia trying to design the next user interface for a cell phone
- He came up with the idea that requirements
  - Should be based on what the users want to do with the phone
- He wrote a use case description for every goal the user has when using the phone
- If the user does not want to do it, it cannot be a use case

# Actors

- A use case captures how an actor uses the system to achieve a goal
- Actors
  - Must be external to the system
  - They are usually human, but not always
  - They often need the system to do something
  - Sometimes, they provide a service for the system

# Actors

- Normally, a use case is started
  - When an actor does something causing the system to do something in response
  - This causes the actor to something else and then the system responds
  - This continues until the goal is reached

# Actors

- Human actors
  - A customer using a web interface
  - An employee using a PC interface
  - A manager requesting a report
- Non-human actors
  - A bank answering queries
  - A credit card company approving transactions
  - A web service providing information

# Eg. Use Case for Visa Requirements

1. The system displays options for investigating visa and vaccination requirements.
2. The user chooses the option to find out about visa requirements.
3. The system prompts the user for the name of the destination country.
4. The user enters the country's name.
5. The system checks that the country is valid.
6. The system prompts the user for her nationality.
7. The user enters her nationality.
8. The system checks the visa requirements of the entered country for a passport holder of her nationality.
9. The system displays the visa requirements.
10. The system displays the option to print out the visa requirements.
11. The user chooses to print the requirements.

Alternative courses:

6. If the country name is invalid:

6.1 The system displays an error message.

6.2 The system returns to step 3.

8. If the nationality is invalid:

8.1 The system displays an error message.

8.2 The system returns to step 6.

9. If no information about visa requirements is found:

9.1 The system displays a suitable message.

9.2 The system returns to step 1.

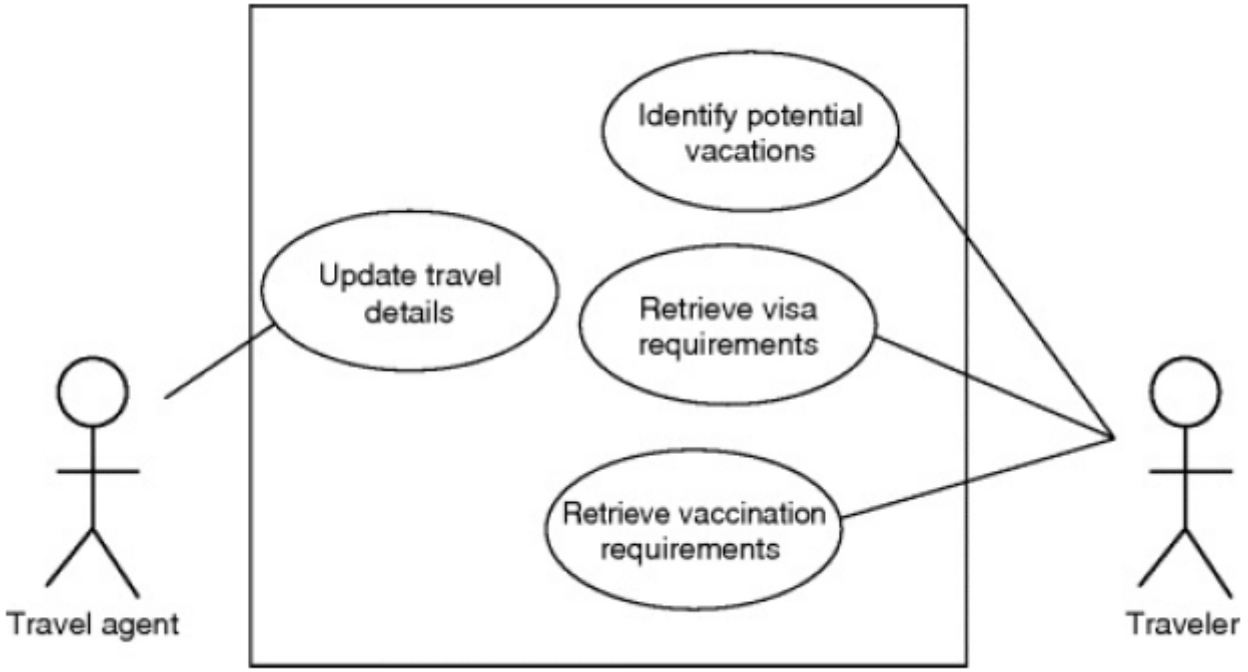
# Writing use cases

- The main series of steps is shown on the left
- The alternate flows are on the right
- each alternate flow describes
  - What happens when something goes wrong
  - How to handle it
  - Where to go after you have handled it
- Steps are numbered
  - Alternate flows start with the number where they can occur in another flow

# Use case diagrams

- High level diagrams which show
  - The users involved
  - The use cases they perform
  - The boundaries of the system as a rectangle
- The details of the individual steps and the information exchanged are in the textual description

# Use case diagrams



# Developing use cases

1. Identify the actors
2. Identify the goal(s) of each actor
3. Write a use case for each goal

# Essential use cases

- Scenarios and use cases have limitations
  - Scenarios are narrowly focused and can obscure wider organizational issues
  - Use cases make assumptions about
    - The system you are interacting with
    - The technology used
    - The type of interaction
- Essential use cases try to avoid these limitations

# Essential use cases

- Essential use cases
  - Try to generalize from specific use cases
  - Avoid the assumptions made in use cases
- They consist of 3 parts
  - A name which expresses the overall intention
  - A series of steps to perform
  - A series of system responsibilities

# Essential use cases

| retrieveVisa<br>USER INTENTION | SYSTEM RESPONSIBILITY |
|---|---|
| find visa requirements | |
| | request destination and nationality |
| supply required information | |
| | obtain appropriate visa information |
| obtain a personal copy of visa information | |
| | offer information in different formats |
| choose suitable format | |
| | provide information in chosen format |

# Essential use cases

- This
  - Reads like a conversation
  - Is high-level and skips the details
  - Is generalized from the use cases