Introduction to Software Engineering – list 4 Use-case specification

Jakub Seredyński 254598

1. Below you have several examples how NOT to define use-case scenarios. Find injected mistakes (some guideline rules were broken) and correct them. One example can contain more than one type of error

Example 1:

Use case: User registration:

1. Asks for user data

2. Enters user data

3. Checks if user data are correct

4. If yes creates an account

Use case: User registration:

(proper order should be 2, 1, …)

1. System asks for user data

   - User data:
   - • First name – not empty text up to 20 characters
   - • Last name – not empty text up to 30 characters
   - • Password – non empty text up to 10 characters, must contain at least 1 letter, 1 digit, 1 special character (!,@,#)

2. User enters user data

3. System verifies if user data are correct

4. System creates an account

Alternative flow (a): User data are not correct

3(a): System informs user that the data is not correct

GOTO step 1 in User registration:

Example 2:

Use case: User registration

1. User selects option Register

2. System asks for user data

3. User enters user data and accepts it with Submit button

Use case: User registration

1. User wants to register

2. System asks for user data

   - User data:
   • First name – not empty text up to 20 characters
   • Last name – not empty text up to 30 characters
   • Password – non empty text up to 10 characters, must contain at least 1 letter, 1 digit, 1 special character (!,@,#)

3. User enters user data

Example 3:

Use case: User registration

1. User wants to register

2. System asks for login

3. User enters login

4. System asks for password

5. User enters password

6. System asks for password repetition

7. User enters password repetition

8. System asks for age

9. User enters age

10. System asks for country

11. User enters country

12. User submits data

Use case: User registration

1. User wants to register

2. System asks for data

    - Data:

        - login
        - password
        - repeated password
        - age
        - country

3. User enters data

4. System verifies data

Alternative flow (a): User data are not correct

5: System informs user that the data is not correct

GOTO step 2 in User registration

2. There is a use-case diagram for a part of on-line library (see Class3). Define a textual specification for selected use-cases. Do not forget about alternative and exceptional flows of events as well as data definitions.

<table>
<tr><td>Use-case name</td><td colspan="3">Sign in</td></tr>
<tr><td>Primary actor</td><td colspan="3">Guest</td></tr>
<tr><td>Brief description</td><td colspan="3">The use-case enables the Guest to login as a Reader or register as a Reader. Only after can the Guest become a Reader or Librarian</td></tr>
<tr><td>Pre-condition</td><td colspan="3">Login page displayed</td></tr>
<tr><td>Post-condition</td><td colspan="3">Reader is logged in</td></tr>
<tr><td rowspan="5">Basic flow of events</td><td>Step.</td><td>Guest</td><td>System</td></tr>
<tr><td>1.</td><td>Wants to sign in</td><td></td></tr>
<tr><td>2.</td><td></td><td>Asks for user data or fingerprint</td></tr>
<tr><td>3.</td><td>Enters user data</td><td></td></tr>
<tr><td>4.</td><td></td><td>Verifies if the user data are valid
Logs Guest in as a Reader or Librarian</td></tr>
<tr><td rowspan="5">Alternative scenario - Guest enters fingerprint</td><td>Step.</td><td>Guest</td><td>System</td></tr>
<tr><td>1a.</td><td>Wants to sign in</td><td></td></tr>
<tr><td>2a.</td><td></td><td>Asks for user data or fingerprint</td></tr>
<tr><td>3a.</td><td>Enters fingerprint</td><td></td></tr>
<tr><td>4a.</td><td></td><td>Verifies if fingerprint is valid
Logs Guest in as a Reader or Librarian</td></tr>
<tr><td rowspan="5">Alternative scenario – Guest doesn’t have an account yet</td><td>Step.</td><td>Guest</td><td>System</td></tr>
<tr><td>1b.</td><td>Wants to Register</td><td></td></tr>
<tr><td>2b.</td><td></td><td>Asks for user data</td></tr>
<tr><td>3b.</td><td>Enters user data</td><td></td></tr>
<tr><td>4b.</td><td></td><td>Verifies if the user data are valid
Creates an User account
GOTO 1</td></tr>
<tr><td rowspan="2">Exception – Guest data incorrect, fingerprint incorrect</td><td>Step.</td><td>Guest</td><td>System</td></tr>
<tr><td>4c.</td><td></td><td>Doesn’t log User as Reader nor Librarian
Asks for user data again
GOTO 2</td></tr>
<tr><td>Data</td><td colspan="3">User data:
- Login – optional if Email, text in ASCII, minimum 4 characters
- Email – optional if Login, text in email format
- Password – obligatory, text in ASCII, minimum 8 characters</td></tr>
</table>

<table>
<tr><td>Use-case name</td><td>Browse catalog</td></tr>
<tr><td>Primary actor</td><td>Reader or Librarian referred to as User</td></tr>
<tr><td>Brief description</td><td>User can browse a catalog, possibly with intention of Reserving a book, on condition that the user isn’t a Librarian</td></tr>
<tr><td>Pre-condition</td><td>Login page displayed</td></tr>
<tr><td>Post-condition</td><td>No changes / book reserved</td></tr>
<tr><td>Basic flow of events</td><td><table>
<tr><td>Step.</td><td>User</td><td>System</td></tr>
<tr><td>1.</td><td>Wants to browse a catalog</td><td></td></tr>
<tr><td>2.</td><td></td><td>Displays the Catalog</td></tr>
</table></td></tr>
<tr><td>Alternative scenario - User is a Reader</td><td><table>
<tr><td>Step.</td><td>Guest</td><td>System</td></tr>
<tr><td>3.</td><td></td><td>Displays “reserve” button next to each position in the Catalog if the book is available</td></tr>
<tr><td>4.</td><td>Chooses to reserve a book</td><td></td></tr>
<tr><td>5.</td><td></td><td>Marks the book as reserved
Assigns the reserved book to User account</td></tr>
</table></td></tr>
<tr><td>Data</td><td>Catalog
- Books – obligatory, list of Book:
- Image - obligatory
- Description – obligatory, text containing book author, title, plot, etc.
- Reserve - optional, button if User is a Reader and book is available
- Available – obligatory, Boolean, not displayed to Reader</td></tr>
</table>

<table>
<tr><td>Use-case name</td><td colspan="3">Manage catalog</td></tr>
<tr><td>Primary actor</td><td colspan="3">Librarian</td></tr>
<tr><td>Brief description</td><td colspan="3">Librarian can add or delete a book from catalog</td></tr>
<tr><td>Pre-condition</td><td colspan="3">Login page displayed</td></tr>
<tr><td>Post-condition</td><td colspan="3">Book catalog changed</td></tr>
<tr><td rowspan="7">Basic flow of events</td><td>Step.</td><td>Librarian</td><td>System</td></tr>
<tr><td>1.</td><td>Wants to manage the catalog</td><td></td></tr>
<tr><td>2.</td><td></td><td>Displays the Catalog</td></tr>
<tr><td>3.</td><td>Wants to add a book</td><td></td></tr>
<tr><td>4.</td><td></td><td>Asks for book data</td></tr>
<tr><td>5.</td><td>Provides book data</td><td></td></tr>
<tr><td>6.</td><td></td><td>Validates the data and adds the book</td></tr>
<tr><td rowspan="3">Alternative flow (a):
Librarian chooses to delete a book</td><td>Step.</td><td>Librarian</td><td>System</td></tr>
<tr><td>3a.</td><td>Chooses a book to delete</td><td></td></tr>
<tr><td>4a.</td><td></td><td>Deletes the book
FINISH</td></tr>
<tr><td rowspan="2">Exception (b):
Book data incorrect</td><td>Step.</td><td>Librarian</td><td>System</td></tr>
<tr><td>6b.</td><td></td><td>Detects that book data is incorrect
Asks for data again
GOTO 5</td></tr>
<tr><td>Data</td><td colspan="3">Catalog
- Books – obligatory, list of Book:
- Image - obligatory
- Description – obligatory, text containing book author, title, plot, etc.
- Delete – obligatory, button, deletes a book from the catalog
- Insert – obligatory, button, adds a book</td></tr>
</table>