

# LBB System

## Use Case: Check-out Books



### Essentials

<b>Use Case UC1:</b>	Check-out Books
<b>Scope:</b>	Library Book Borrowing System
<b>Level:</b>	User Goal
<b>Goal in Context:</b>	The intention of the User is to check-out books from a library. Only one User can check-out books at any one time.
<b>Actors:</b>	Primary:
	- User (becomes Member once s/he has identified him/herself with the System)
	Secondary:-
<b>Preconditions:</b>	Request books are available and loan system is in service.
<b>Postconditions:</b>	User has checked out the requested books.
<b>Failed End Condition:</b>	-
<b>Trigger:</b>	User wants to check-out some books.
<b>Main Scenario:</b>	<ol style="list-style-type: none"> <li>1. User requests system to check-out books.</li> <li>2. User <u>identifies him/herself</u> to System.</li> <li>3. Member <u>registers book</u> with System.</li> </ol> <p><i>Step 3 is repeated for each book that is checked-out by Member.</i></p> <ol style="list-style-type: none"> <li>4. Member indicates to System that s/he has finished checking out books.</li> <li>5. System records all books registered by Member as on loan, requests Printer to print out a receipt* for the session, and puts itself in a state to receive the next User.</li> </ol>
<b>Extensions:</b>	<p>2a. User fails to <u>identifies him/herself</u> with System: use case ends in failures.</p> <p>3a. System informs Member that s/he has reached his/her maximum number of books allowed on loan; use case continues at step 4</p>

	<p>3b. Member requests System to remove book from the books that are checked out:</p> <p>3b.1. Member identifies book to System.</p> <p>3b.2a. System identifies book and removes it from the list of books registered by Member; use case continues from where it was interrupted.</p> <p>3b.2b. System fails to identify book; use case continues from where it was interrupted.</p> <p>(3-4)a. Member requests System to cancel check-out.</p> <p>(3-4)a.1. System removes all books that were registered by Member, and puts itself in a state to receive the next User; use case ends in failure.</p> <p>(3-4)b. System times-out waiting for input from Member:</p> <p>(3-4)b.1. System removes all books that were registered by Member, and puts itself in a state to receive the next User; use case ends in failure.</p>
Open Issues:	* <i>&lt;list of issues awaiting decision affecting this use case&gt;</i>

## Related Information

*<this section contains additional information for a more detailed and complete use case description. This information is usually added in a third or fourth description iteration>*

Special Requirements:	* <i>&lt;list additional or detailed technological or data requirements either for the whole use case or for individual steps. In the later case reference the step number from the above scenario descriptions.&gt;</i>
Priority:	* <i>&lt;how critical to your system / organization, use the standard categories use in your organizations&gt;</i>
Performance:	* <i>&lt;The amount this use case or a single step should take.&gt;</i>
Frequency:	* <i>&lt;how often it is expected to happen&gt;</i>
Due Date:	* <i>&lt;date or release needed&gt;</i>
Super Use Cases:	* <i>&lt;name of use case(s) that includes this one&gt;</i>
Sub Use Cases:	* <i>&lt;links to sub use cases&gt;</i>

\* The use case description is not complete; it is filled out only as far as needed for the course.

Derived from "RE-A2 Requirements Engineering with Use Cases", S. Sendal & A. Strohmeier, [SWEED](#), 2001

## [Description](#) | [Case Studies](#)

---

Last modified: 15.08.2002 , [M.Kropp](#)