

Internet Engineering

Tomasz Babczyński, Zofia Kruczkiewicz
Tomasz Kubik

Information systems modelling– UML and service description languages

Laboratory 2

Choose yourself and new technologies



Wrocław University of Technology



Project co-financed from the EU European Social Fund



Design patterns used to build the Service Subtier of the BusinessTier

D.Alur, J.Crupi, D. Malks, Core J2EE. Desin Patterns

Outline of creating the Library Catalogue Java Application

1. Description of an additional process/processes of Business based on the example from laboratory 1
2. Download and open the initial UML project Library2_UML2.vpp as the Visual Paradigm 12.00 CE project
3. The formulation of additional functional requirements and non-functional system
4. Model analysis of the entire system based on these additional use case/cases on the use case diagram
5. Extension of the design model of the business service sub-tier based on the extended class diagram and sequence diagrams, created by iteratively driven development of use cases after addition the new uses case/cases.
6. Implementation of the additional business service/services of sub-tier created in a series of iteration driven development project model



1. Description of Business – proposal of an extension of the example

1. Description of human resources

- An library employee may add new titles to the catalog of titles. Each title is represented by the following data: title, author, publisher, ISBN, and the number of copies and their storage place and is present in the library as a single piece of information for each title.
- Some books are recorded on the tape, so their data of a title also includes additional data, eg name of the actor.
- Each piece, whether it is a book or a cassette is described in a separate copy number, and also data indicating (it applies to discrete units) information on the number of days on which you can borrow a copy.
- The numbers of book can be repeated among books of different titles.
- A librarian can add new titles and copies and search them, and the customer can only browse titles and accessible copies of selected titles.
- Realization of loan of books based on assumption that library has a set of many clients. Each client can loan to 5 books of different titles. Time of the loan of the book based on type of books - the books without period have the standard time of the loan and books with the given period of the loan will start at current date and will finish after this period.

2. Provisions

- The employee is responsible for the accuracy of the data - is responsible for material non-compliance with the state of the rental.

3. Technical data

- The customer can view data of library through the website or directly through a special program. A library employee can also insert, modify and delete data of titles and copies. It is assumed that customers, searching at the same time the catalogue of the library, may be more than 1,000 and the library can contain some of thousands of titles and at least twice as many copies. The library consists of several centers in various cities across the country (list of cities is included in the contract). It is recommend to use Java technology.



2. List of requirements of the catalog titles and books – an extension of list

2.1.functional

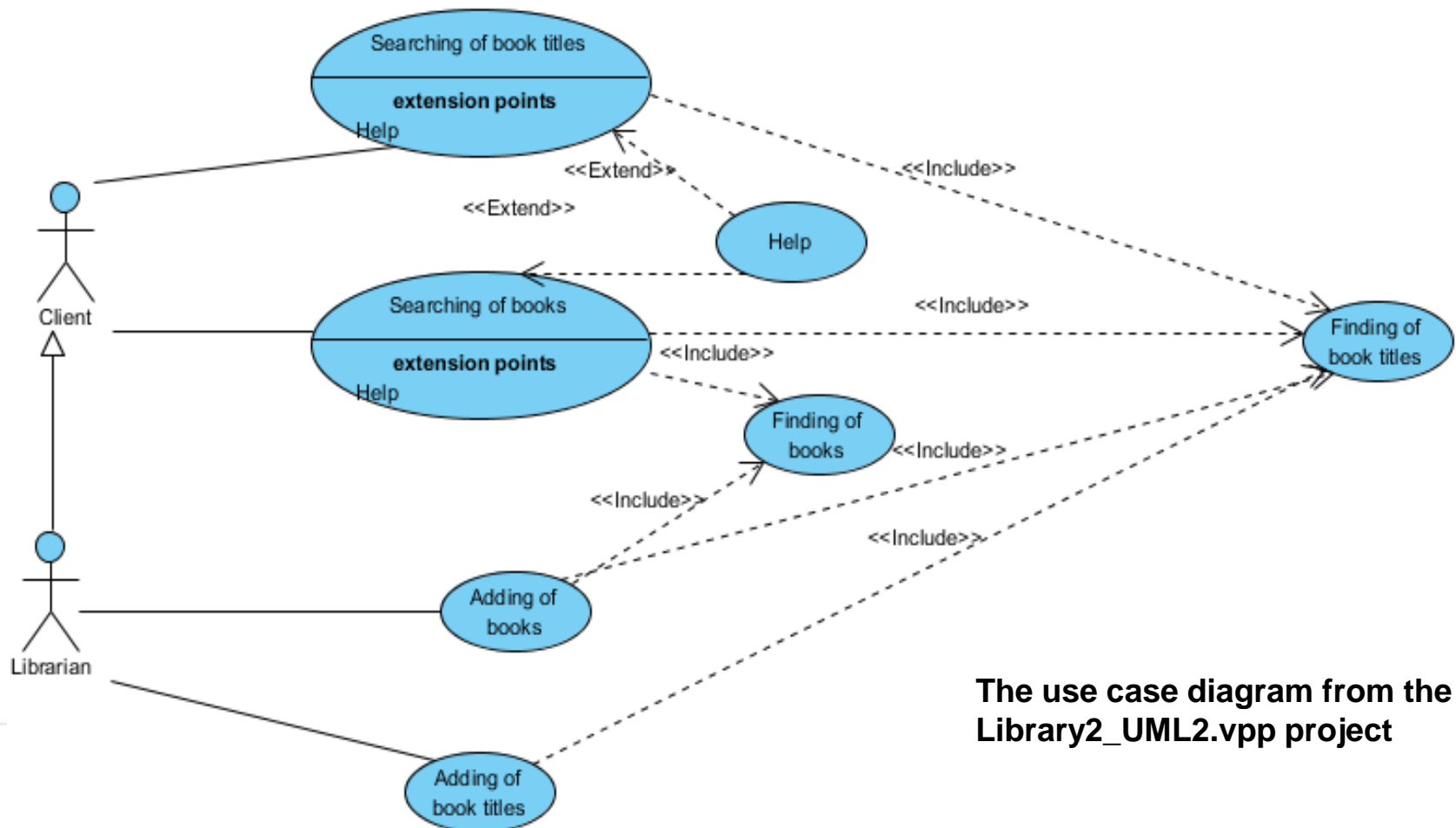
- The system includes a catalog of titles
- The system includes two types of copies such as books and cassettes with recordings of audio books.
- Each copy contains the title, author name, ISBN, publisher, if this is the book, plus the name of the actor, if it is a sound recording.
- It can be a lot of copies of books and tapes with the same title. Each copy of the book or the cassette has the nonrepeating number of ISBN or the pair data such as ISBN, and the actor's name among all copies.
- In order to select the correct copy, has to give the ISBN, if it is a book and also the name of the actor, when it is a cassette
- Both copies of a book or a cassette, can be spent on borrow at the obligatory period and for a specific period

2.2. nonfunctional

- Inserting data of titles and their copies may be made only by authorized persons
- Searching for information can be done by the client
- Operations management and information retrieval can be made via the Internet or by an application that runs without browser.



3. Use Case Diagram of Library Catalogue - it must extend the use case diagram by addition of a new use case/cases by using the Visual Paradigm tool (the link to the Website of the actual version of the tool of the Visual Paradigm International Company, is in the section of Tools of the page of laboratory instructions.)



The use case diagram from the
Library2_UML2.vpp project



Actor	Description	Use cases
Librarian	<i>The librarian is responsible for maintaining the resource of the library (inserting and deleting: titles of books, copies of books). It may also browse the directory resources of titles and copies of books</i>	<ul style="list-style-type: none">• Adding of book titles• Adding of book• Searching of titles• Searching of books• <u>Loan of books</u>
Client	<i>The client can only search the resources of the catalogue of titles and books</i>	<ul style="list-style-type: none">• Searching of titles• Searching of books• <u>Loan of books</u>





UC Finding of book titles

DESCRIPTION

GOAL: Searching of a title

PC (preconditions): initialization by running the program (such as opening a web page, start the application)

FC (final conditions) It gets the new title, or provides information about your lack of result

Scenario

1. Searching for the proceeds according to attributes: ISBN (mandatory) and actor (if required) according to the data given to the use case
2. If there is a title of a given attributes, it returns this title, otherwise it is returned information for lack of a title.

UC Searching of book titles

DESCRIPTION

GOAL: Search titles

PC (preconditions): initialization by running the program (such as opening a web page, start the application)

FC (final conditions) searching titles with the given value of a mandatory attribute ISBN or ISBN, and actor in the case of a cassette with the audio book, or information regarding the lack of result

Scenario:

1. It must be specified the attributes of the title: ISBN as the mandatory value plus actor, if it is looking for a cassette with the audio book. It creates a standard title to search for the real one.
2. It must call **the Finding of book titles use case** to make sure that the title of the specified attributes already exists. If not, the use case finishes without information about the title, otherwise it delivers the title.



UC Finding of books

DESCRIPTION

GOAL: Looking for a book

PC (preconditions): initialization by running the program (such as opening a web page, start the application)

FC (final conditions) providing a book containing the same data as the pattern book or providing information about the lack of the book

SCENARIO:

1. Finding a copy of the proceeds according to the attributes: the number of the book (obligatory) and according to the title given to the use case. It is searching books of the given title.
2. If it finds a book of the specified number, it returns the existing one, otherwise it is returned information of lack of a book.

UC Searching of books

DESCRIPTION

GOAL: Searching for copies of the book with the given title

PC (preconditions): initialization by running the program (such as opening a web page, start the application)

FC (final conditions): It finds a book of the title consistent with the mandatory ISBN attribute, or ISBN and the actor in the case of audio book and the mandatory number of a copy or gives information about the lack of copy

SCENARIO:

1. Please specify the attributes title: ISBN plus actor, if you are looking for the title of the book as the audio book. It creates a standard title to search the real one.
2. It calls **the Finding of book titles use case** to make sure that the title of the specified attributes already exists. If not, it completes the use case not giving information about the title, otherwise it returns the searched title.
3. You must create a pattern of a book which contains the number of the searched book and then you must call **the Finding a book use case**, providing the pattern book. The result given by the executed use case should be given as a final score.



UC Adding of book titles

DESCRIPTION

GOAL: Inserts a new title

PC (preconditions): initialization by running the program (such as opening a web page, start the application)

FC (final conditions): It adds a title which includes the following mandatory attributes: title, author, ISBN, publisher, and if it is an audio book, the actor's name, or information about the existence of such a title

SCENARIO:

1. Provide attributes of the title : title, author, ISBN, publisher, and if it is an audio book, the name of the actor. It creates a title for searching and for the possible insertion.
2. It executes **the Finding od book titles use case** giving the new book title. It makes sure that the title of the specified attribute already exists. If so, it must just finish the use case, otherwise you must insert a new title.

UC Adding of books

DESCRIPTION

GOAL: Adding a new book

PC (preconditions): initialization by running the program (such as opening a web page, start the application)

FC (final conditions): It inserts a book consistent with the mandatory attribute ISBN, or ISBN and actor in the case of the audio book and the specified number of the book and possibly the attribute to specifying the date of return, or give information about the existence of such a copy

SCENARIO:

1. It specifies the attributes of the title: ISBN as the mandatory plus actor, if you are looking for is the title of the book as audio book. It creates a pattern title to searching for the real one.
2. It executes **the Finding of book titles use case** providing the pattern title. It makes sure that the title with the specified attribute already exists. If not, it finishes the use case not giving information about the title.
3. Otherwise you must create a book that contains the given number and the date attribute of return, if required, and must pass it to **the Finding of books use case**. If there is not a book with a given number, it adds this book, otherwise you must return the existence of such a copy.



It must extend the number of definition of a new use case/cases

UC Loan of books

DESCRIPTION

GOAL:

PC (preconditions):

FC (final conditions):

SCENARIO:





4. An analysis of commonality and variability - it must extend the number of classes based on the use case diagram with a new use case/cases

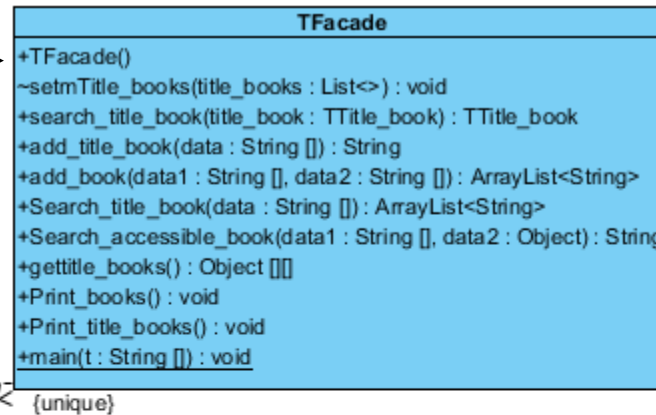
- Detected two main classes of "Entity" for liability reasons: **the TTitle_book class** (it includes attributes of the title, has books -it adds and searches for them) and **the TBook class** (a number). The concepts of the book and the copy of the book are equivalent.
- Inheritance has been detected in the properties of titles that may occur as a simple book or an audio book (**TTitle_book_on_tape class** as the type "Entity", which inherits from **the TTitle_book class**). It is defined a strategy for the storage of the title of multiple copies of books or tapes. **The TBook** object can be borrowed for the standard period and the audio books as **the TBook_period** can be borrowed for specified period, in the period attribute.
- The relationship between **the TTitle_book** and **TBook** objects are in relation 1 to 0 .*. The compound objects **TTitle_book** inherit from **TTitle_book_on_tape** the same relationship 1 to 0 .*. From **the TBook class** are inherited **TBook_period class**. Hence the ordinary books can be identified only by numbers or numbers and the date of repayment. This also applies to books in the form of sound recordings.
- Aggregation detects as the strong relationships between the title and a copy - a copy can not exist without title. It can be chosen **the strategy pattern** for the implementation of **TBook** objects.
- **The TFacade class** uses a "Control" as a facade pattern to separate objects of type "Entity" from the rest of the class system and **the TFactory class** as the "Control" object to create different types of titles and copies and to separate the creation and using of objects
- The identification process of new classes based on scenarios of additional uses case/cases taking into account the appropriate creational, structural and behavioral design patterns. It is the main requirement of the assessment of this laboratory - to use the proper design patterns of each kind of standard design patterns. You may use the Visual Paradigm for UML Community Edition for creation the class diagram and sequence diagrams- the link to the Website of the actual version of the tool of the Visual Paradigm International Company, is in the section of Tools of the page of laboratory instructions.



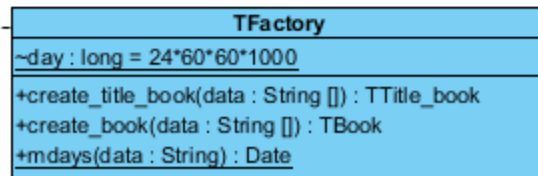
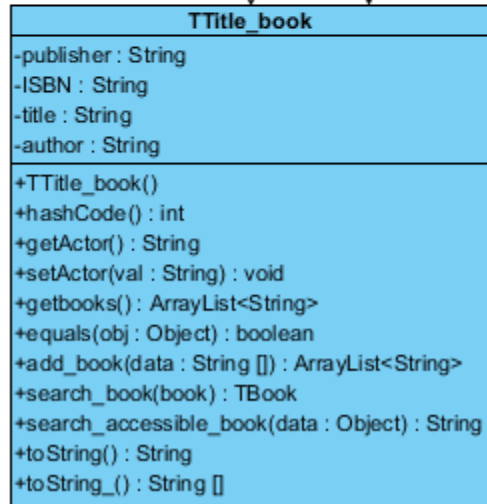
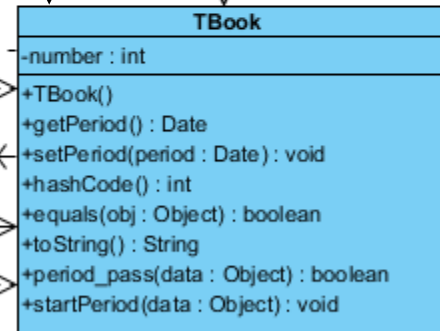
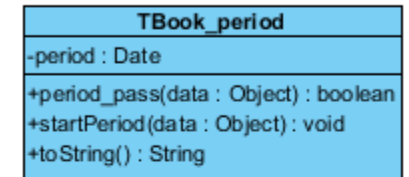


The class diagram from the
Library2_UML2.vpp project

Facade



Strategy



Factory

It must extend the class
diagram by new class
/classes

Flyweight





Code of a new class/classes as the skeleton/skeletons





Fifth Iteration

Make the sequence diagram and code of the
new use case/cases based on the instruction
of the laboratory 1





Sequence diagram/diagrams





Wrocław University of Technology

Master programmes in English
at Wrocław University of Technology



The fifth iteration - code



HUMAN CAPITAL
HUMAN – BEST INVESTMENT



Wrocław University of Technology

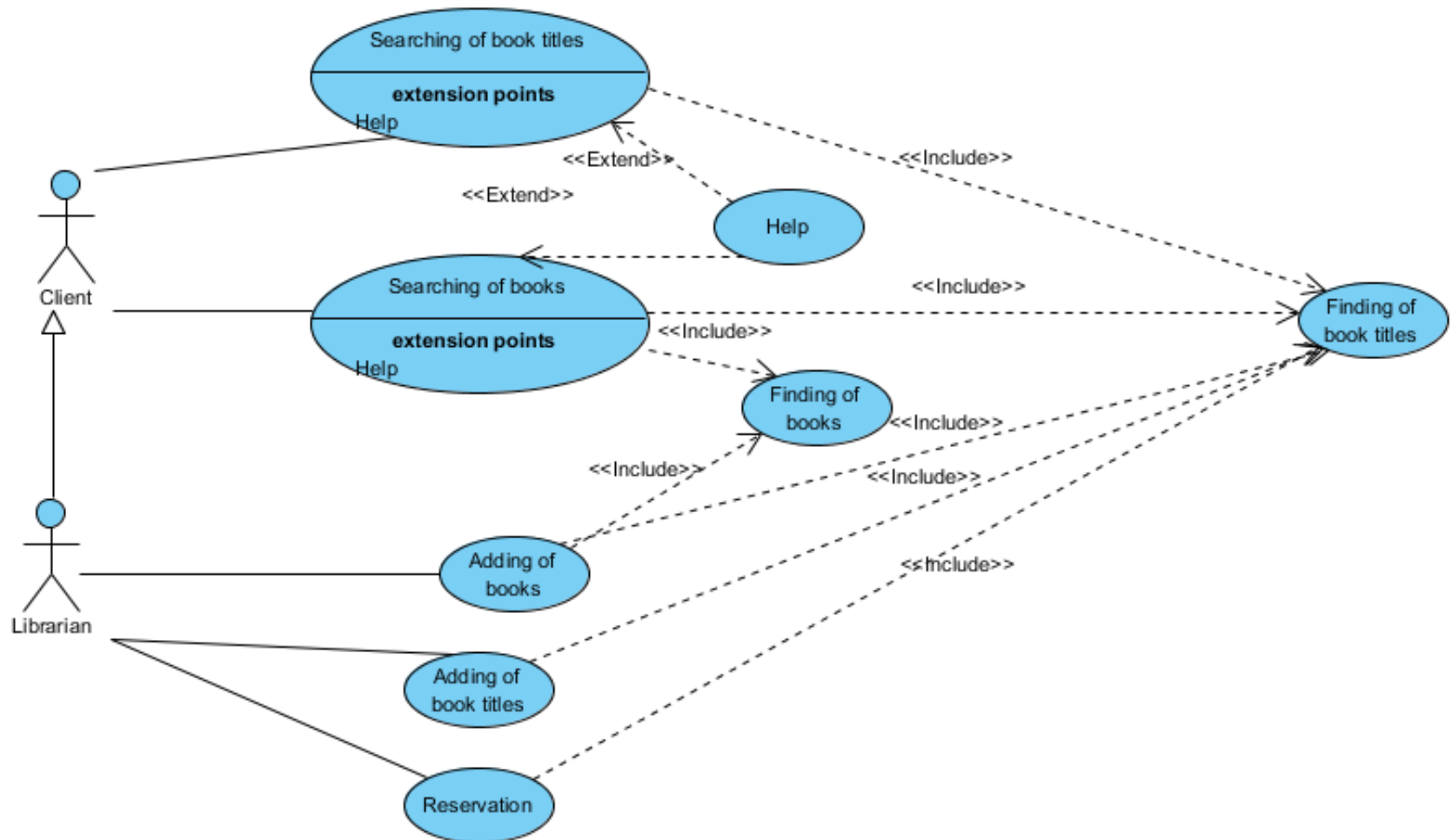
EUROPEAN
SOCIAL FUND



Project co-financed from the EU European Social Fund

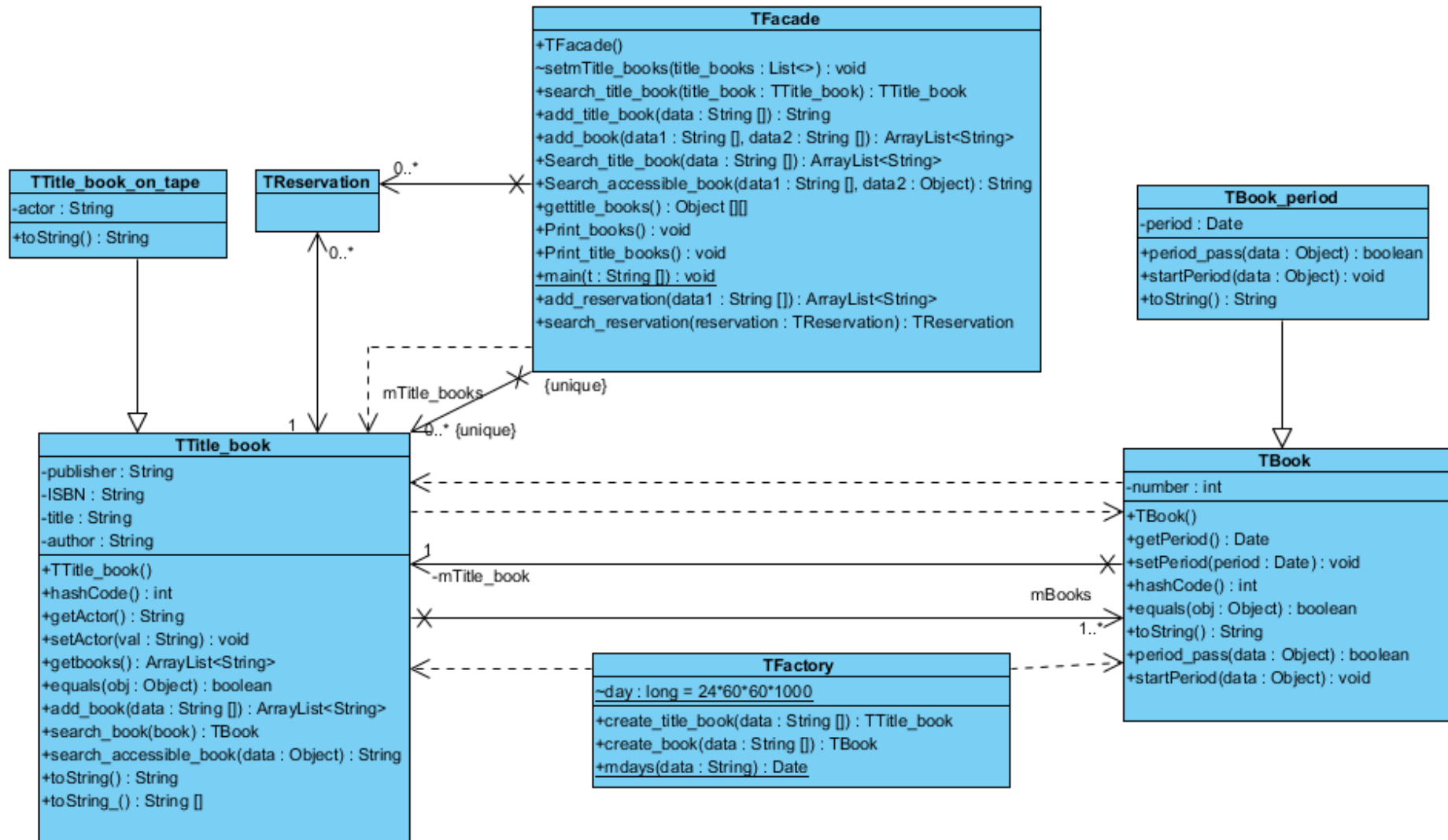


Supplement 1 (assessment: 4.0)– Library2_UML3.vpp project



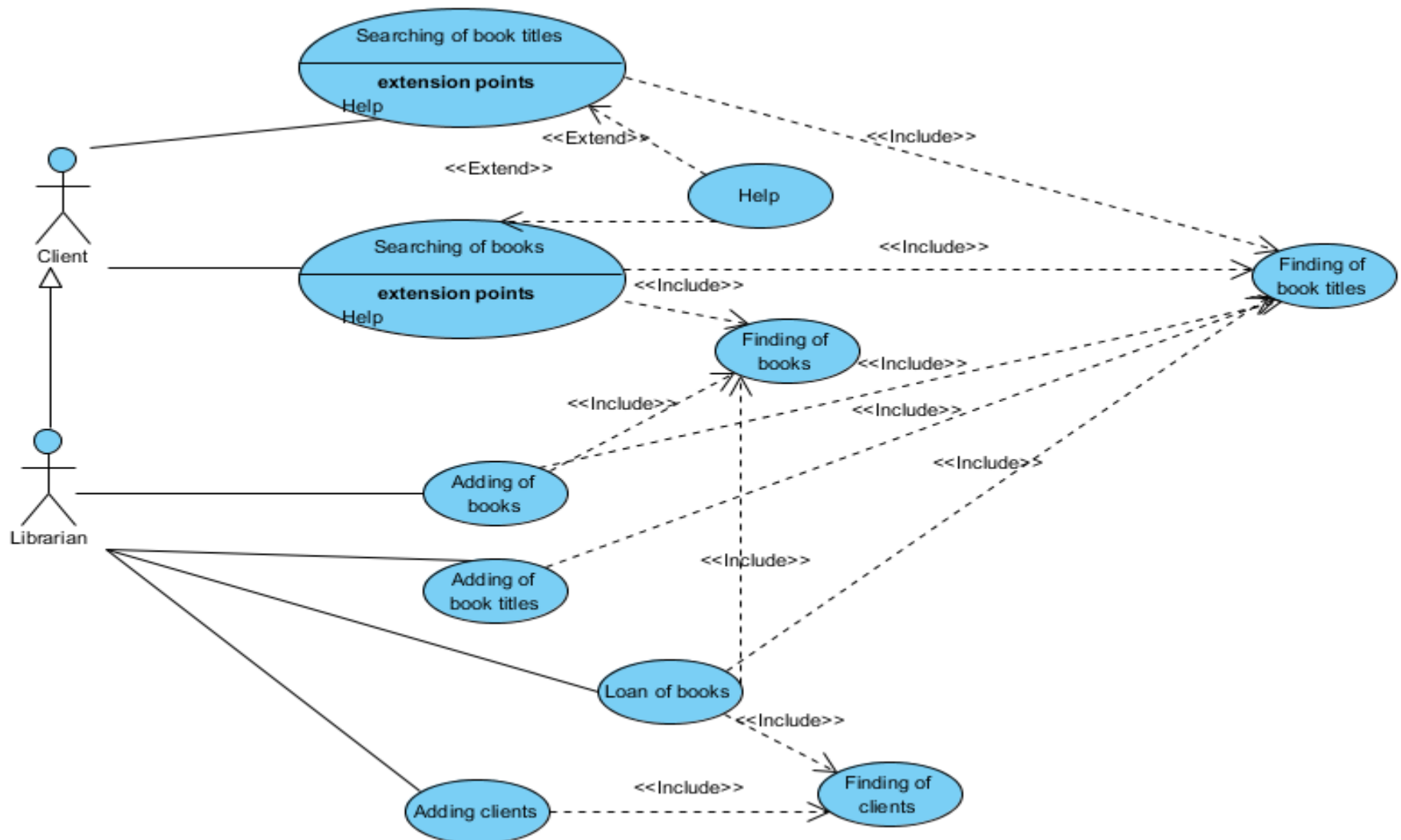


Supplement 1(continuation) – Library2_UML3.vpp project





Supplement 2 (assessment: 5.0-5.5) – Library2_UML4.vpp project





Supplement 2 (continuation) – Library2_UML4.vpp project

