

UML Activity Diagram

Formative Exercise with Feedback

This is a practice assignment, to help you learn UML. You will have a couple of days to complete each practice assignment, after which the solutions will be revealed on MOLE, so that you can check your attempt. **This is not the assessed assignment.**

Read the scenario carefully, and then create a **UML activity diagram** that faithfully captures the information requested in the instructions after the scenario. The objective of this exercise is to assess *how accurately* you can develop a structured model from unstructured requirements. There is an exact answer.

You may use any drawing package, or CASE tool, to develop the diagram, so long as you conform to the UML 2 standard notation (avoid non-standard, or out of date UML tools).

Scenario: The Applied Science Library

The Applied Science Library needs a new computer system to manage the circulation of books and periodicals in the library. The librarian decides how the annual budget may be spent. Technical services first acquire the books and periodicals from the publisher and then catalogue them on a separate database. After this, reader services manage the circulation system. Their activity consists of issuing, discharging and renewing loans to borrowers and calculating when loans are overdue. To be issued with a loan or renew a loan, a borrower must present a UCard to the reader services clerk with the item to be loaned. A scanner is used to scan both the barcode on the item and the barcode on the UCard, after which the loan may be issued. To discharge a loan, the borrower need only present the item, which is scanned, before the loan is discharged. If any barcode scan fails, then the task cannot be completed. A borrower can ask to reserve an item, or cancel a reservation. If a book or periodical on loan has been reserved by another borrower, then it cannot be renewed. Overdue loans are calculated daily by reader services. For each loan that has fallen overdue, a message is sent by email to the borrower. If a loan is overdue when it is discharged, the borrower must pay a fine, before the loan can be discharged. If a borrower has any overdue loans, no further loan can be issued or renewed to that borrower.

In phase 1, the new circulation system will allow reader services to handle all circulation-related tasks from computer terminals on the front desk, including issuing, discharging and renewing loans, and determining when loans are overdue. In phase 2, the extended system will also allow borrowers to reserve books and cancel reservations, using other computer terminals in the library.

Activity Diagram

Create an activity diagram modelling the part of the *circulation system* that deals specifically with *lending*, that is, issuing, discharging and renewing loans, and paying any fines. Do not model making reservations. You should aim to capture the *control flow* (ignoring data flow) of the lending process. Assume the following:

- The first things to model are how the borrower presents an item and optionally a UCard to the reader services clerk.
- The last things to model are how the reader services clerk performs some action and optionally returns the UCard to the borrower.
- The reader services clerk has a choice of actions to take, depending on what the borrower has requested.
- The diagram will model scanning, issuing, discharging, renewing and payment of fines (according to the scenario).
- When fines are due, these can always be paid (there is no choice required here).
- The diagram will model phase 2 of the system, in which it is possible for items to be reserved (but do not model reservation).

Before constructing your diagram:

- Identify the different branching flows in the scenario, which will be modelled as flows leaving decision points.
- Identify the mutually exclusive and exhaustive guards that govern each branching flow.
- Identify where alternative flows are merged again, which will be modelled as flows reaching merge points.
- Identify the entry and exit points, and remember to mark success and failure exit points suitably. Don't try to execute multiple scenarios in one pass, but let scenarios fail, if they are prevented.
- Sketch the activity diagram, linking the flows from the entry point to each relevant activity, decision point or merge point, leading eventually either to a success or failure exit point.
- If your control flows have to cross, please use a little semicircular "hop" over the crossed line.
- Keep your diagram as simple as possible – check whether you could share common actions on one path, before branching.