

Smart Library Request Workflow in ServiceNow

Problem Statement

- In many academic institutions, managing a library system manually creates challenges such as lack of real-time updates on book availability, inefficient approval processes, and limited visibility into borrowing patterns. Students often face issues in knowing which books are available, while librarians struggle with tracking requests, approving or rejecting them, and maintaining accurate records of issued or returned books.
- To address these challenges, a role-based Library Management System needs to be implemented on the ServiceNow platform. This system should allow students to request books, while librarians manage the book inventory and approve borrow requests. Automated workflows, role-based access controls, and reports will ensure a streamlined process for both parties.

Objective

The objective of this project is to implement a role-based Library Management System in ServiceNow that simplifies the process of borrowing and managing books. The system allows students to submit borrow requests and track their status, while librarians manage the book catalog and approve or reject requests. Through automated workflows, book availability is updated in real time, role-based access ensures proper security, and reporting features like “Most Borrowed Books” provide insights for better resource management.

Phase 1: Requirement Analysis & Planning

The system requires two roles: student and librarian. Students can request and track books, while librarians manage the catalog and approve requests. Two tables are needed: **Book** (details, status) and Borrow Request (transactions). Workflows will automate approvals and status updates, ACLs will control access, and reports like “Most Borrowed Books” will give insights. Planning covers defining tables, roles, workflows, and test cases to ensure smooth implementation.

Business Objectives

The main business objective of this project is to create a digital library system in ServiceNow that reduces manual effort, improves accuracy in book tracking, and enhances user experience for both students and librarians. By automating borrow requests, approvals, and book status updates, the system ensures efficient operations, transparency, and faster access to resources.

Additionally, role-based controls and reporting features support better decision-making and effective management of library resources.

Functional Scope

The Library Management System on ServiceNow will provide the following key functions:

- **Book Management** – Librarians can add, update, and maintain book details with status tracking (Available, Issued, Lost, Returned).
- **Borrow Requests** – Students can create and track requests; librarians can approve, reject, or close them.
- **Workflow Automation** – Approval of requests will automatically update the book status.
- **Role-Based Access** – Students have limited access to requests, while librarians manage the entire system.
- **Reporting** – Generate insights such as “Most Borrowed Books” for library usage analysis.

Stakeholder Mapping

- **Students** – Request and track books; expect easy access and quick approvals.
- **Librarians** – Manage catalog, approve/reject requests, generate reports; need efficiency and accuracy.
- **System Administrators** – Configure tables, workflows, and roles; ensure secure and stable systems.
- **Management/Faculty** – Use reports for planning and resource allocation.
- **IT Support Team** – Provide technical support, upgrades, and system reliability.

Execution Roadmap

1. **Role Setup** – Create student and librarian roles.
2. **Table Design** – Build *Book* and *Borrow Request* tables with required fields.
3. **Form Configuration** – Add fields, set default values, and establish related lists.
4. **Workflow Automation** – Configure Flow Designer for approval and book status updates.
5. **Access Control** – Define ACLs for role-based permissions.
6. **UI Enhancements** – Apply UI Policies for form restrictions and usability.
7. **Reporting** – Create dashboards/reports like *Most Borrowed Books*.
8. **Testing & Validation** – Use sample data, impersonation, and debug tools to verify end-to-end flow.

Milestone 1: Create Roles

- Define roles: **student** and **librarian**.
- Ensure role separation for access control (students = request, librarians = manage).

Milestone 2: Create Tables

- Create a Book table (u_book).
- Create a Borrow Request table (u_borrow_request).
- Link Borrow Request → Book.

Milestone 3: Add Fields to Book

- Add fields: Title, Author, ISBN, Status.
- Configure Status choices: Available (default), Issued, Lost.

Milestone 4: Add Fields to Borrow Request

- Add fields: Requested By, Book (Reference), Request Date, Status.

- Configure Status choices: Requested (default), Approved, Rejected, Returned.

Milestone 5: Show Borrow Requests on Book (Related List)

- Add Borrow Request related list on Book form.
- Verify relationship: Book → Borrow Request.

Milestone 6: Prevent Requesting Already-Issued Books (Reference Qualifier)

- Configure Reference Qualifier on Borrow Request → Book field.
- Condition: Show only books where Status = Available.

Milestone 7: Flow Designer – Approve → Issue Book

- Create Flow: Trigger when Borrow Request → Status changes to Approved.
- Action: Update Book → Status = Issued.
- Optional: Send email notification to requester.

Milestone 8: UI Policy

- Apply UI Policy: When Borrow Request Status = Approved → lock fields (Book, Requested By, etc.).
- Improves form usability and prevents accidental edits.

Milestone 9: Access Control (ACLs)

- Define ACLs for **Book table**:
 - Librarians = Create/Write/Delete.
 - Students & Librarians = Read.
- Define ACLs for **Borrow Request table**:
 - Students = Create/Read (own records).

- Librarians = Approve/Write/Delete.

Milestone 10: Report – “Most Borrowed Books”

- Build Report from Borrow Request table.
- Group by **Book**, Count requests.
- Display the top 5 most borrowed books as a chart or list.

Phase 2: Backend Development & Configurations

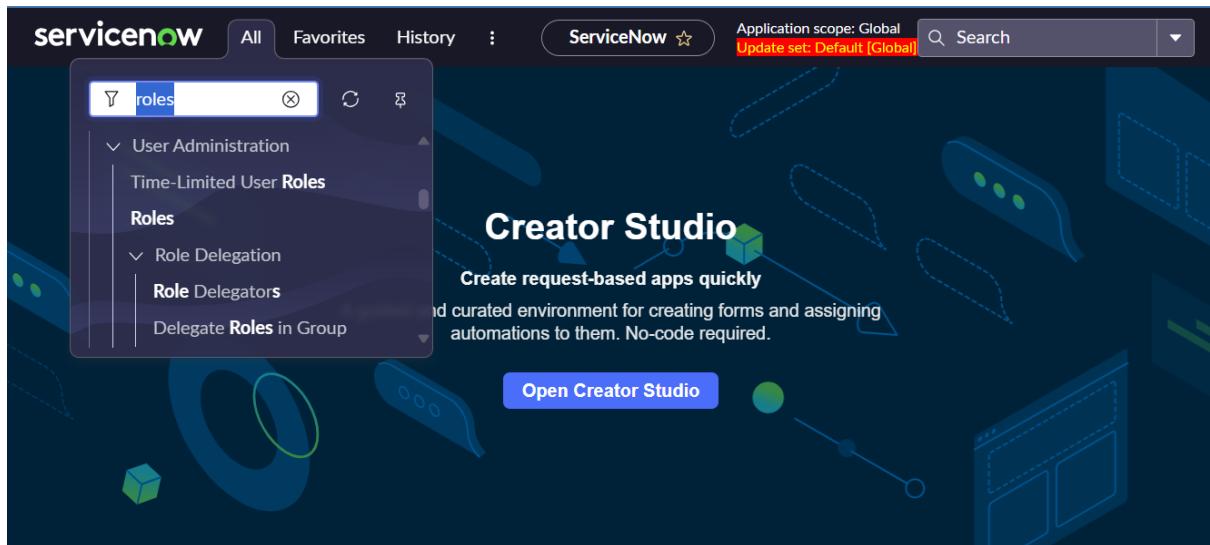
The backend setup focuses on creating roles, tables, and automation to manage the library process efficiently. Roles such as *student* and *librarian* are defined to control access, while custom tables (*Book* and *Borrow Request*) are configured with fields and relationships to store catalog and transaction data. A reference qualifier ensures only available books can be requested, and Flow Designer automates approvals by updating book status to *Issued*. To maintain consistency, UI Policies lock fields after approval, and Access Control Rules (ACRs) enforce role-based permissions. Finally, reports like *Most Borrowed Books* are configured to provide administrators with useful insights.

Data Architecture

Creation of roles

Steps to Create Roles

- Open ServiceNow
- Click on All → search for Roles.
- Select User Administration → Roles.
- Click on New.
- Enter Role Name as *librarian* → click Submit.
- Again, click on New.
- Enter Role Name as *student* → click Submit.
- Verify that both roles (*librarian* and *student*) are listed in the Roles module.



This screenshot shows a list view of roles. The top navigation bar includes 'Roles' in the title and a 'New' button highlighted with a red box. The list table has columns for Name, Description, Elevated privilege, and Updated. Several rows are listed, such as 'u_ticket_request_user', 'u_metro_ticket_user', and 'librarian'.

	Name	Description	Elevated privilege	Updated
	u_ticket_request_user		false	2025-08-20 22:40:38
	u_metro_ticket_user		false	2025-08-20 22:28:50
	u_borrow_request_user		false	2025-08-19 02:15:59
	u_book_user		false	2025-08-19 02:15:26
	student		false	2025-08-19 02:13:45
	librarian		false	2025-08-19 02:13:22

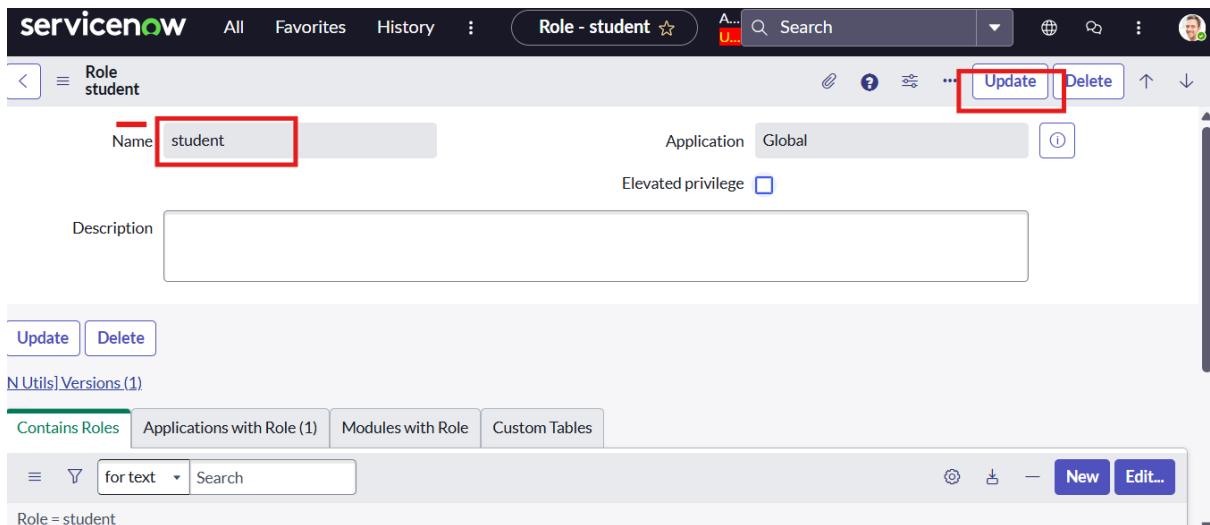
This screenshot shows the details view for the 'librarian' role. The top navigation bar includes the role name 'Role - librarian'. The main form has fields for Name ('librarian'), Application ('Global'), and Elevated privilege (checkbox). Below the form is a 'Description' text area and a footer with 'Update' and 'Delete' buttons, both of which are highlighted with red boxes.

SN Utils] Versions (1)

Contains Roles Applications with Role (1) Modules with Role Custom Tables

for text Search

New Edit...



Roles					
	Name	Description	Elevated privilege	Updated	
	u_metro_ticket_user		false	2025-08-20 22:28:50	
	u_borrow_request_user		false	2025-08-19 02:15:59	
	u_book_user		false	2025-08-19 02:15:26	
	student		false	2025-08-19 02:13:45	
	librarian		false	2025-08-19 02:12:29	
	service_api_admin		false	2025-06-25 23:38:16	

Steps to Create Tables

1. Open ServiceNow.
2. Click on All → search for Tables.
3. Select System Definition → Tables.
4. Click on New.
5. Enter Label as *Book* → click Submit (this creates the table u_book).
6. Again, click on New.
7. Enter Label as *Borrow Request* → click Submit (this creates the table u_borrow_request).
8. Verify that both tables (*Book* and *Borrow Request*) are listed under the Tables module.

Servicenow Roles list page. A context menu is open over a row, with the 'Tables' option highlighted by a red box.

Role	Elevated privilege	Updated
student	false	2025-08-20 22:28:50
librarian	false	2025-08-19 02:15:59
service_api_admin	false	2025-08-19 02:15:26
	false	2025-08-19 02:13:45
	false	2025-08-19 02:12:29
	false	2025-06-25 23:38:16

Servicenow Table - book configuration page. The 'Label' field ('book') and 'Name' field ('u_book') are highlighted by a red box.

A table is a collection of records in the database. Each record corresponds to a row in a table, and each field on a record corresponds to a column on that table. Applications use tables and records to manage data and processes. [More Info](#)

Column	Type	Reference	Max length	Default value	Display
Title	String	(empty)	40		true
Class	System Class Name	(empty)	80	javascript:current.getTableName();	false
status	Choice	(empty)	40		false
Author	String	(empty)	40		false
ISBN	String	(empty)	40		false
Sys ID	Sys ID (GUID)	(empty)	32		false
Created by	String	(empty)	40		false
Created	Date/Time	(empty)	40		false
Updated by	String	(empty)	40		false

Servicenow Table - book configuration page. The 'Table Columns' table is shown, with the 'Column label' column header and the first five rows ('Titel', 'Class', 'status', 'Author', 'ISBN') highlighted by a red box.

Column label	Type	Reference	Max length	Default value	Display
Titel	String	(empty)	40		true
Class	System Class Name	(empty)	80	javascript:current.getTableName();	false
status	Choice	(empty)	40		false
Author	String	(empty)	40		false
ISBN	String	(empty)	40		false
Sys ID	Sys ID (GUID)	(empty)	32		false
Created by	String	(empty)	40		false
Created	Date/Time	(empty)	40		false
Updated by	String	(empty)	40		false

The screenshot shows the ServiceNow 'Table - Borrow Request' configuration page. At the top, there's a navigation bar with 'servicenow', 'All', 'Favorites', and a search bar. Below the title, it says 'Table - Borrow Request' with a star icon. The main area has a blue header bar with 'Delete', 'Update' (which is highlighted with a red box), and 'Delete All Records' buttons. A tooltip explains what a table is. The table definition section shows two fields: '* Label' set to 'Borrow Request' and '* Name' set to 'u_borrow_request'. To the right, it says 'Application Global' and 'Remote Table'. Below this, there are tabs for 'Columns', 'Controls', and 'Application Access', with 'Columns' selected. The 'Table Columns' list shows one item: 'for text'. There's a search bar and a toolbar with various icons. At the bottom, there's a pagination area showing '1 to 11 of 11' and a 'New' button.

Steps to Add Fields to Book

1. Open ServiceNow.
2. Click on All → search for Tables.
3. Select System Definition → Tables.
4. From the list, open the table Book (u_book).
5. Scroll down to the Columns (Dictionary) related list.
6. Click on New to add a new field.
7. Enter Column Label as *Title*, select *Type = String → click Submit.
8. Again, click on New. Enter Column Label as *Author*, select *Type = String → click Submit.
9. Again, click on New. Enter Column Label as *ISBN*, select *Type = String → click Submit.
10. Again, click on New. Enter Column Label as *Status*, select *Type = Choice → click Submit.
11. Open the Status field you just created.
12. In the Choices related list, click New and add values: *Available*, *Issued*, *Lost*.
13. Set the Default Value of Status = *Available*.
14. Click Update to save.
15. Verify that the Book form now has fields: *Title*, *Author*, *ISBN*, *Status*.

The screenshot shows the ServiceNow Table - book configuration page. At the top, there's a navigation bar with 'servicenow' and links for 'All', 'Favorites', 'History', and a search bar labeled 'Table - book'. Below the navigation is a toolbar with actions: 'Delete', 'Update' (which is highlighted with a red box), and 'Delete All Records'. A 'Columns' tab is also highlighted with a red box. The main area shows a table of columns with the following data:

Column label	Type	Reference	Max length	Default value	Display
Titel	String	(empty)	40	true	
Class	System Class Name	(empty)	80	javascript:current.getTableName();	false
status	Choice	(empty)	40		false
Author	String	(empty)	40		false
ISBN	String	(empty)	40		false
Sys ID	Sys ID (GUID)	(empty)	32		false

A 'New' button is located in the top right corner of the table area.

The screenshot shows the ServiceNow Dictionary Entry - status configuration page. At the top, there's a navigation bar with 'servicenow' and links for 'All', 'Favorites', and a search bar labeled 'Dictionary Entry - status'. Below the navigation is a toolbar with actions: 'Create Choice List', 'Delete Column', 'Update' (highlighted with a red box), and a 'New' button. The main area shows a table of choices with the following data:

Label	Value	Language	Sequence	Inactive	Updated
Available	A	en		false	2025-08-19 02:30:20
Issued	I	en		false	2025-08-19 02:30:57
Lost	L	en		false	2025-08-19 02:31:22

An 'Actions on selected rows...' dropdown is visible at the top right of the table area. A 'Choices (3)' tab is highlighted with a red box.

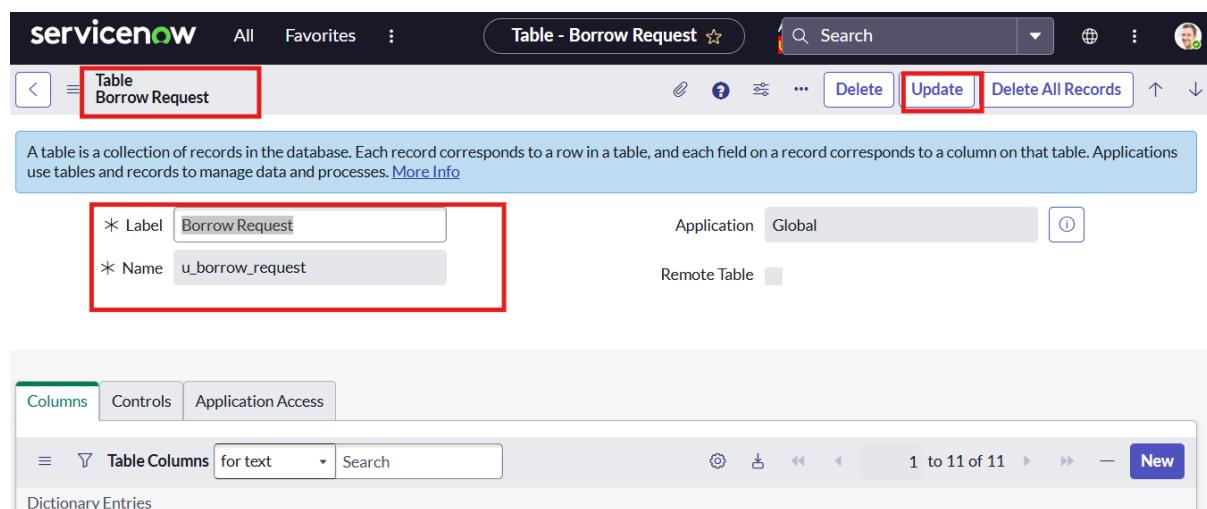
The screenshot shows the ServiceNow book - New Record creation screen. At the top, there's a navigation bar with 'servicenow' and links for 'All', 'Favorites', and a search bar labeled 'book - New Record'. Below the navigation is a toolbar with actions: 'Submit' (highlighted with a red box). The main area shows a form with fields: 'Author' (text input), 'Titel' (text input), 'ISBN' (text input), and 'status' (dropdown menu). The 'status' dropdown is open, showing options: '-- None --', 'Available', 'Issued', and 'Lost'. The 'Available' option is highlighted with a blue selection bar.

Steps to Add Fields to Borrow Request

1. Open ServiceNow.
2. Click on All → search for Tables.

3. Select System Definition → Tables.
4. From the list, open the table Borrow Request (u_borrow_request).
5. Scroll down to the Columns (Dictionary) related list.
6. Click on New to add a new field.

7. Enter Column Label as *Requested By*, select Type = Reference, set Reference = User [sys_user] → click Submit.
8. Again, click on New. Enter Column Label as *Book*, select Type = Reference, set Reference = Book [u_book] → click Submit.
9. Again, click on New. Enter Column Label as *Request Date*, select *Type = Date/Time → click Submit.
10. Again, click on New. Enter Column Label as *Status*, select *Type = Choice → click Submit.
11. Open the Status field you just created.
12. In the Choices related list, click New and add values: *Requested, Approved, Rejected, Returned*.
13. Set the Default Value of Status = *Requested*.
14. Click Update to save.
15. Verify that the Borrow Request form now has fields: *Requested By, Book, Request Date, Status*.



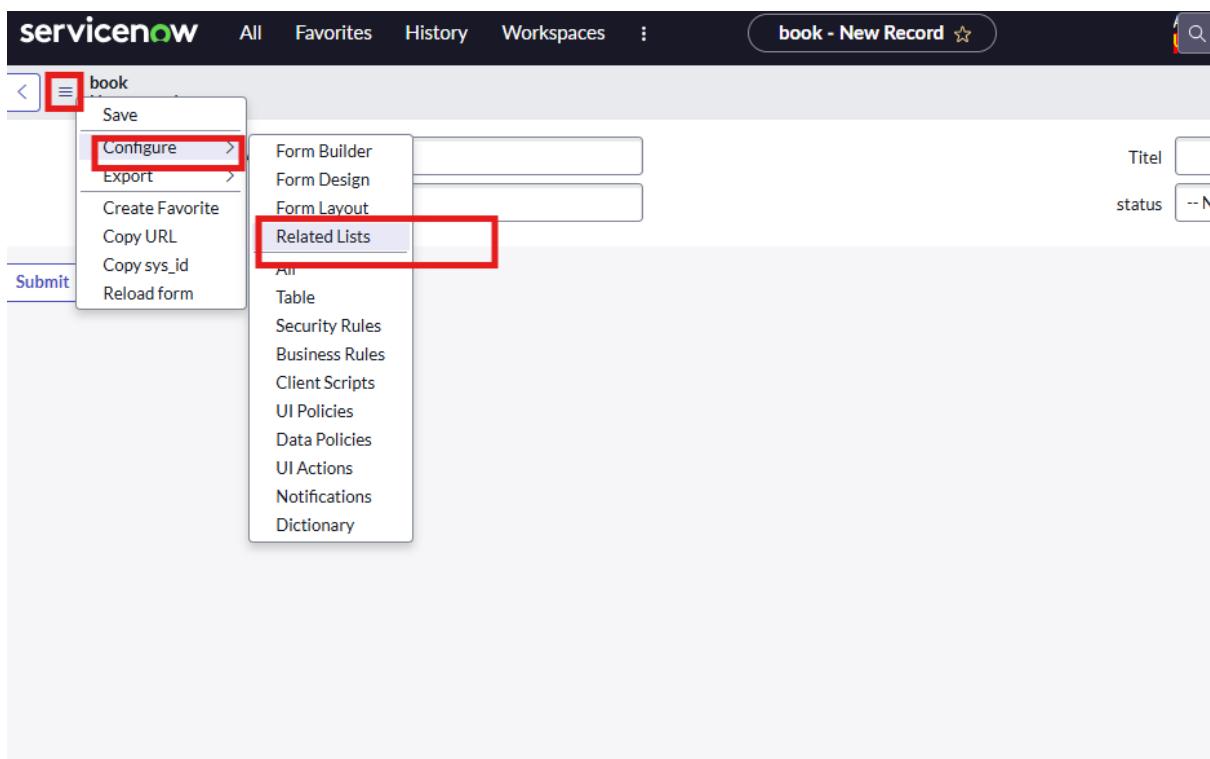
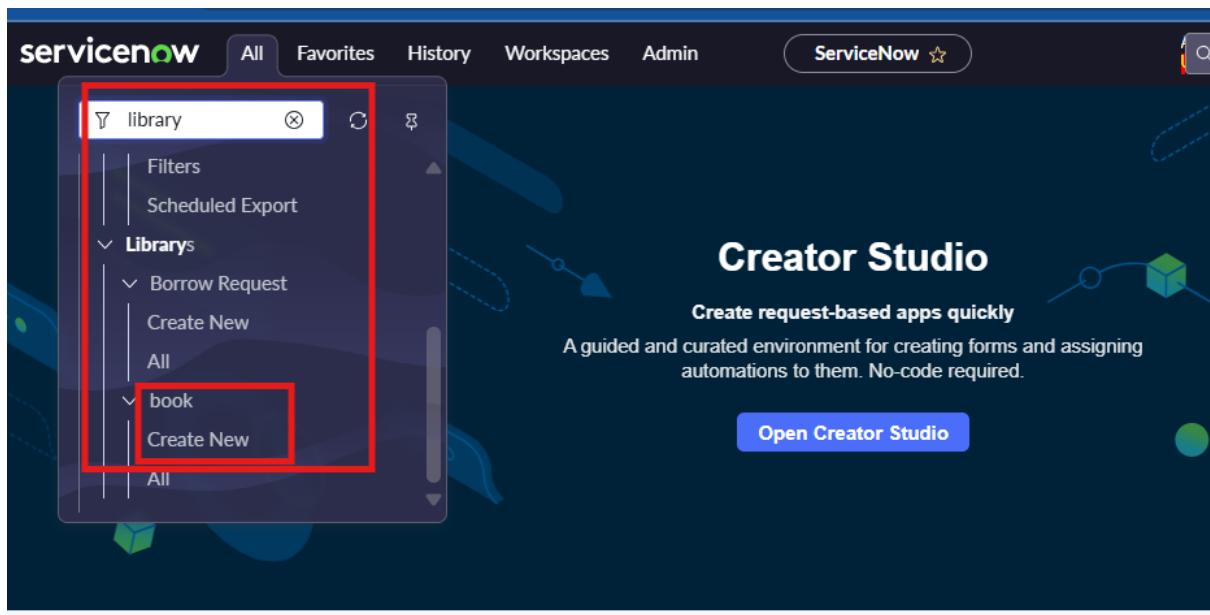
The screenshot shows the ServiceNow interface for managing tables. The top navigation bar includes 'servicenow', 'All', 'Favorites', and a search bar labeled 'Table - Borrow Request'. Below the header, there's a toolbar with 'Delete', 'Update' (which is highlighted with a red box), and 'Delete All Records'. A message box provides information about tables and records. The main area displays the 'Borrow Request' table settings. A red box highlights the 'Label' field containing 'Borrow Request' and the 'Name' field containing 'u_borrow_request'. To the right, application details show 'Application: Global' and 'Remote Table:'. At the bottom, a 'Columns' tab is selected, showing a table of columns with a 'New' button. The table has columns for 'Table Columns' (set to 'for text'), 'Search', and navigation buttons.

The screenshot shows the ServiceNow interface for the 'Borrow Request' table. The top navigation bar includes 'All', 'Favorites', 'History', 'Workspaces', and a search bar. Below the title 'Table - Borrow Request' are tabs for 'Columns', 'Controls', and 'Application Access'. The main area displays a list of columns with their properties: Column label, Type, Reference, Max length, Default value, and Display. A red box highlights the 'Book' column, which is a Reference type pointing to the 'Book' table. The 'Display' column for this row is set to 'false'. At the bottom of the list is a button labeled 'Insert a new row...'. The top right corner of the window has buttons for 'Delete', 'Update', 'Delete All Records', and 'New'.

The screenshot shows the ServiceNow interface for the 'Status' dictionary entry. The top navigation bar includes 'All', 'Favorites', 'History', 'Workspaces', and a search bar. Below the title 'Dictionary Entry - Status' are tabs for 'Create Choice List', 'Delete Column', and 'Update'. The main area displays a list of choices with columns: Label, Value, Language, Sequence, Inactive, and Updated. A red box highlights the 'Approved' choice, which has a value of 3 and a language of en. The 'Update' button at the top right is also highlighted with a red box. The bottom right corner of the window has a button labeled 'Actions on selected rows...'.

Steps to Configure Reference Qualifier (Prevent Requesting Already-Issued Books)

1. Open ServiceNow.
2. Click on All → search for Tables.
3. Select System Definition → Tables.
4. From the list, open the table Borrow Request (u_borrow_request).
5. Scroll down to the Columns (Dictionary) related list.
6. Open the field Book (Reference field to Book [u_book]).
7. In the field form, scroll to the Advanced View (click the gear if needed).
8. Locate the Reference Qualifier section.
9. Select Simple type.
10. In the Reference Qualifier Condition, set: Status is Available
11. Click Update to save the field.



servicenow All Favorites History Workspaces Admin ServiceNow Search Cancel Save

Configuring related lists on book form

Available: Attachments
Selected: Borrow Request->Book

> < ▲ ▼

Cancel Save

View name: Default view

servicenow All Favorites History Workspaces Dictionary Entry - Book Search Delete Column Update

Dictionary Entry Book

* Column name u_book Read only

Mandatory Display

Reference Specification Choice List Specification Default Value

The Reference field specifies what table this field displays values from.

Reference book

Reference qual condition Add Filter Condition Add OR Clause

status is Available AND OR X

Delete Column Update

Related Links Show Table Run Point Scan

servicenow All Favorites History Workspaces Borrow Request - Create Created Search

Borrow Request New record

Requested By Book Request Date

Status -- None --

email Approved Rejected Requested Returned

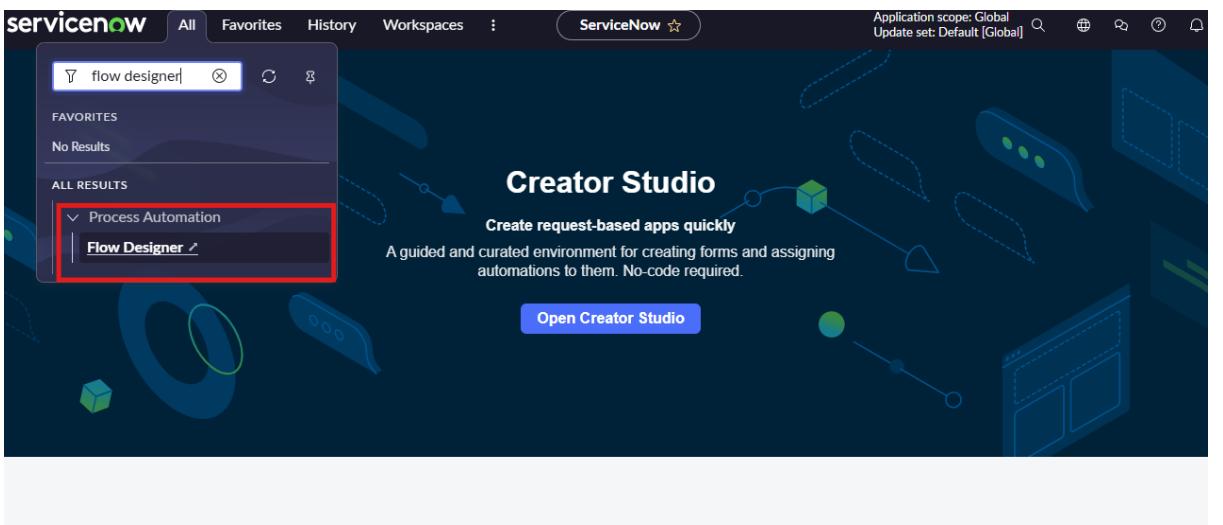
Submit

Phase 3: Development

Steps to Create Flow in Flow Designer (Approve → Issue Book)

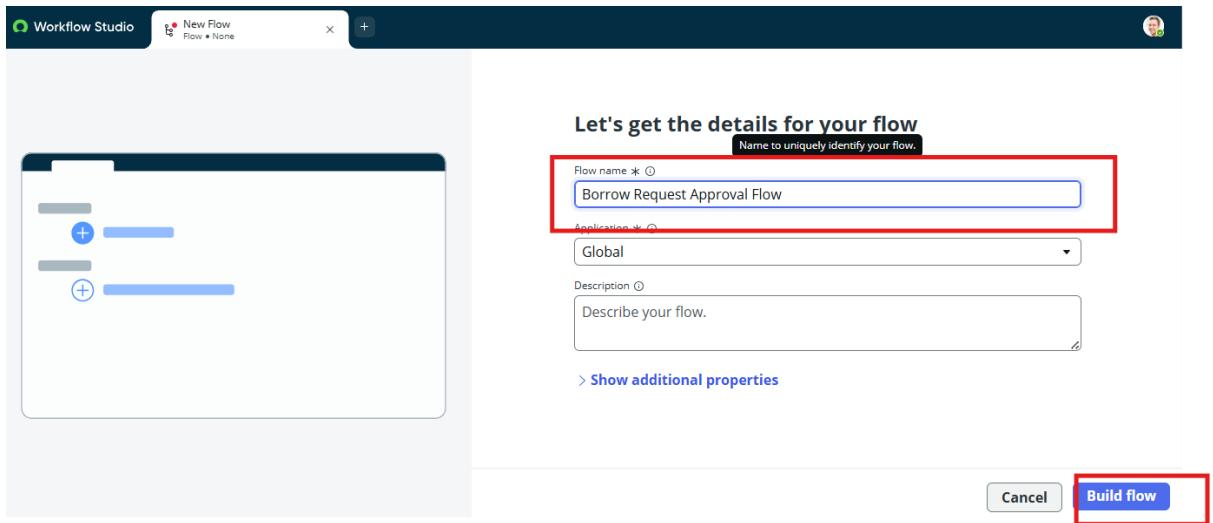
1. Open Flow Designer

1. Open ServiceNow.
2. Click on All → search for Flow Designer.
3. Select Flow Designer and click New → Flow.
4. Enter Name as Borrow Request Approval Flow → click Submit.



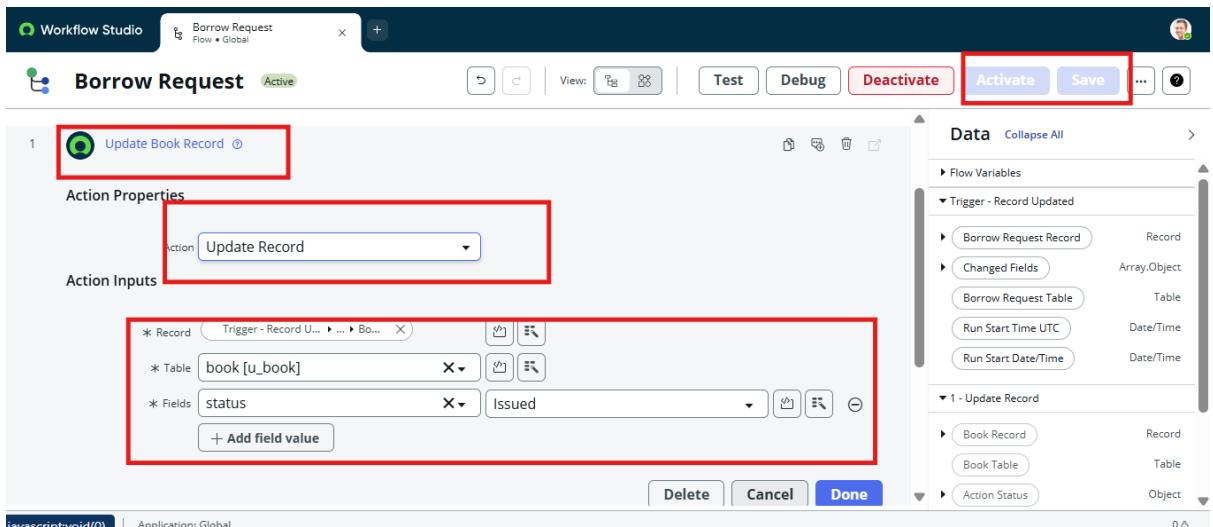
The screenshot shows the ServiceNow Workflow Studio interface. The title bar says "Workflow Studio". Below it, a message box says "New version available: Upgrade to get the latest Workflow Studio features." The main area shows a table of flows. The first column has a checkbox, the second column is "Name", the third is "Application", the fourth is "Status", and the fifth is "Active". The rows show flows like "Borrow Request" (Global, Published, true), "form Request" (Global, Draft, false), etc. At the top of the flow list, there is a "New" button with a dropdown menu. The "Flow" option in this menu is highlighted with a red box. To the right of the table, there are sections for "Latest updates" and "Recent activity".

	Name	Application	Status	Active
	Borrow Request	Global	Published	true
	form Request	Global	Draft	false
	Set visibility of Analytics applications(v2)	Global	Published	true
	New Application Admin Task State Updated	Creator Studio	Published	true
	New Application Task State Updated	Creator Studio	Published	true
	New Request App Approval Flow	Creator Studio	Published	true
	Collaboration Request Flow	Collaboration Request	Pending	true



2. Add Trigger (When to Run the Flow)

1. In the Flow Designer canvas, click + Add Trigger.
2. Select Created or Updated.
3. Choose Table = Borrow Request [u_borrow_request].
4. Add condition: Status = Requested.
5. Click Done.



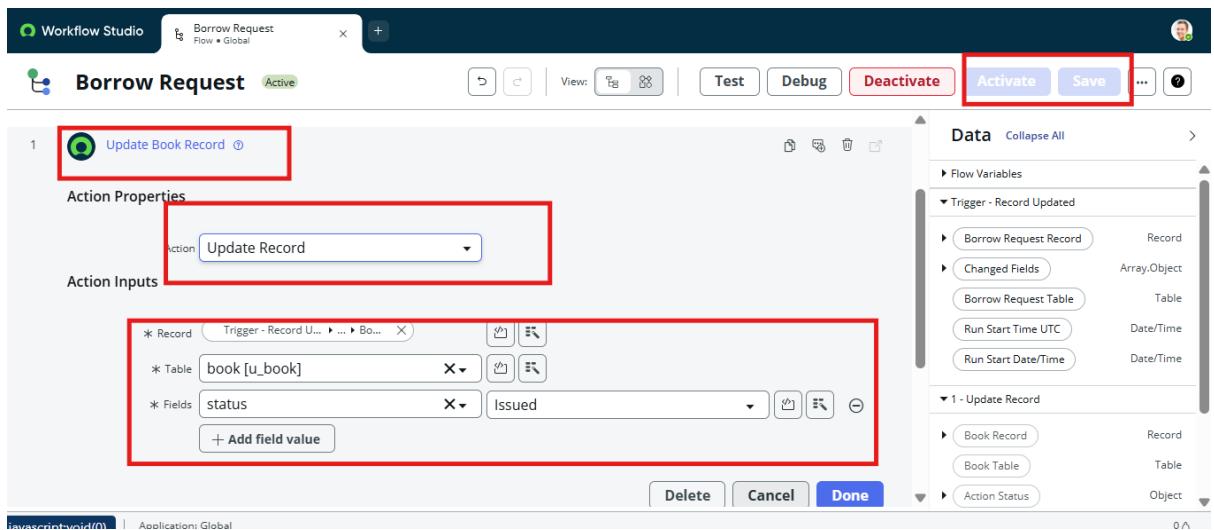
3. Add Action – Approval

1. Click on + Add Action.
2. Search for Ask for Approval.
3. Set Approvers = Role: librarian.

4. Approval conditions:
On Approve → continue flow.
On Reject → set Status = Rejected.
5. Click Done.

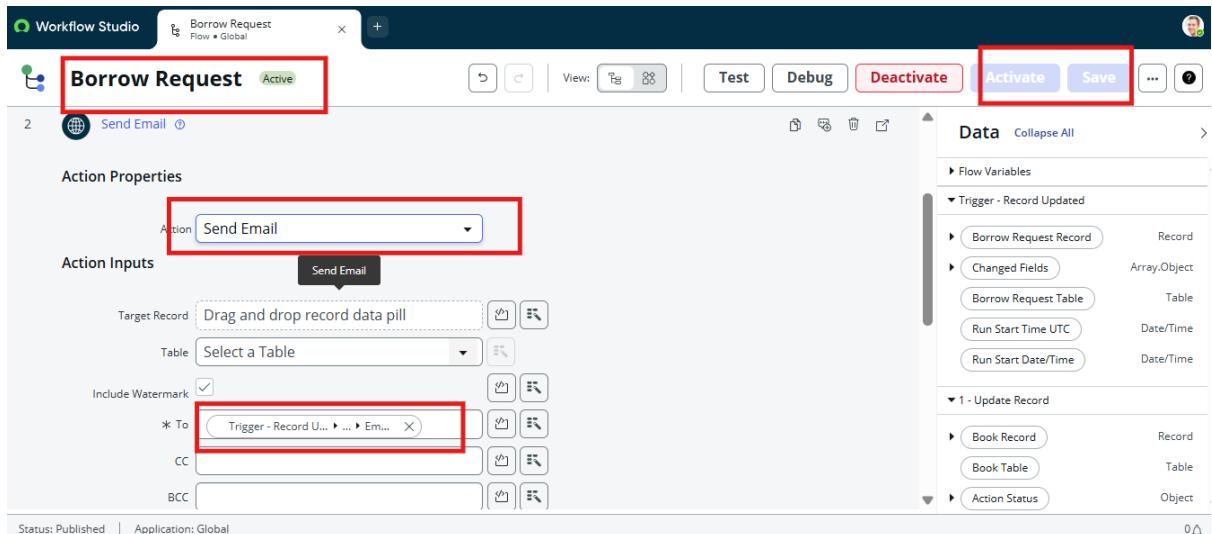
4. Add Action – Update Record (Issue Book)

1. Click on + Add Action.
2. Search for an Update Record.
3. Choose Table = Book [u_book].
4. Condition: Sys ID = Current.Book (the one from Borrow Request).
5. Set Status = Issued.
6. Click Done.



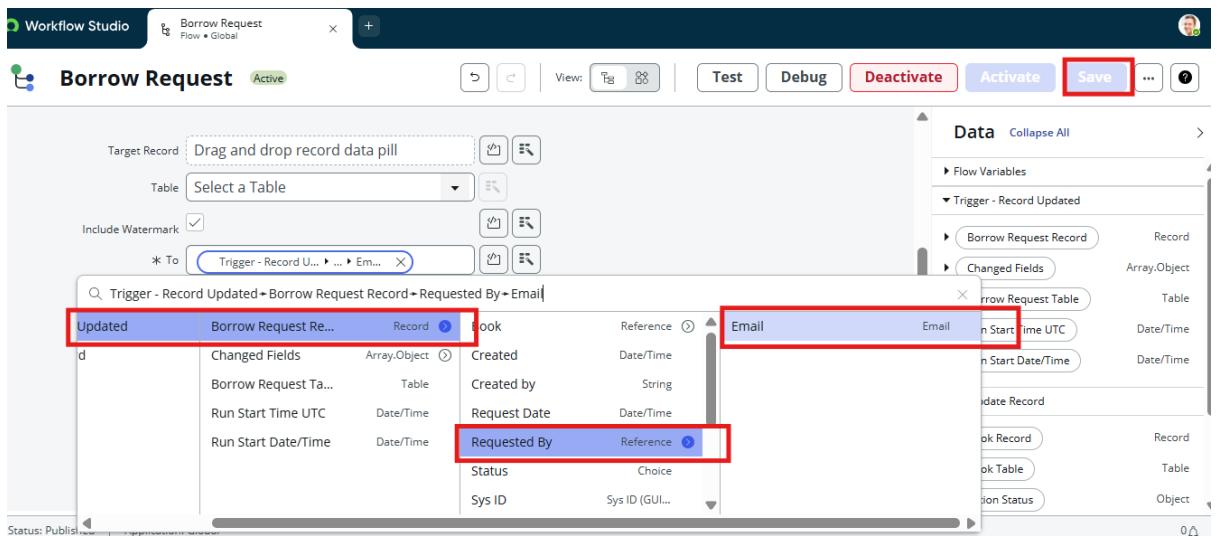
5. Add Action – Update Borrow Request Record

1. Click on + Add Action.
2. Search for an Update Record.
3. Choose Table = Borrow Request [u_borrow_request].
4. Condition: Sys ID = Current.
5. Set Status = Approved.
6. Click Done.



6. Add Action – Send Email (Notification)

1. Click on + Add Action.
2. Search for Send Email.
3. Set To = Requested By (Borrow Request → Requested By field).
4. Subject: Your Borrow Request has been Approved.
5. Body: The book [Book.Title] has been issued to you. Please collect it from the library.
6. Click Done



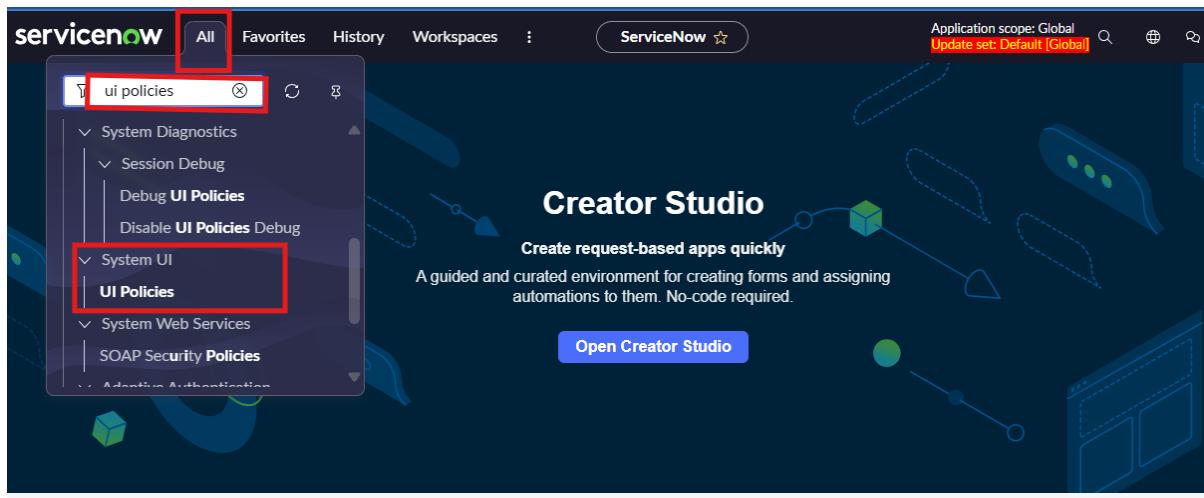
7. Save and Test Flow

1. Click Save → Activate.
2. Create a test Borrow Request with Status = Requested.

- Verify: Librarian gets approval → On approve, Book.Status = Issued, Borrow Request.Status = Approved, and student receives an email.

Steps to Create UI Policy

- Open ServiceNow.
- Click on All → search for UI Policies.
- Select System UI → UI Policies.



- Click on New.
- Enter Table = Borrow Request [u_borrow_request].
- Enter Short Description = Make Return Date mandatory when Issued.
- Submit the record.

UI policies change fields on a form based on a set of conditions. Use UI policies to show or hide fields, or to make fields mandatory or read-only based on these conditions. A UI policy specifies one or more actions to take when the policy is triggered. First, create the policy. Then add as many actions as necessary. UI policy actions are applied only if all of the following conditions are met:

- The UI Policy is Active
- The items in the Conditions field evaluate to true
- The field specified in the UI policy action is present on the specified form

[More info](#)

Table	Borrow Request [u_borrow_request]	Application	Global
Short description		Make Return Date mandatory when Issued.	
Order	100		
When to Apply	Script		
Conditions	Add Filter Condition	Add OR Clause	
-- choose field -- -- operator -- -- value --			

- Reopen the created UI Policy record.
- In the Conditions section, set: Status = Issued.

10. In the UI Policy Actions related list, click New.

The screenshot shows the 'UI Policy - Lock after approval' page in ServiceNow. At the top, there's a condition: 'Status is Approved'. Below this, under 'UI Policy Actions (2)', there's a new row being created for 'Field name: u_book'. The row has columns: Field name, Mandatory, Visible, and Read only. The 'Mandatory' column is set to 'Leave alone' and the 'Visible' column is set to 'Leave alone'. Both 'Read only' and 'Visible' are set to 'True'.

Field name	Mandatory	Visible	Read only
u_book	Leave alone	Leave alone	True
u requested by	Leave alone	Leave alone	True

11. Select Field = Return Date.
12. Set Mandatory = True.
13. Set Visible = True.
14. Submit the UI Policy Action.

Phase 4: Data Migration, Testing & Security

Access Control

Access Control for Book Table

1. Open ServiceNow.
2. Click on All → search for Access Control.
3. Select System Security → Access Control (ACL).
4. Click New.
5. Select Table = Book [u_book].
6. Choose Operation = read.
7. Set Requires role = student, librarian.

servicenow All Favorites History Workspaces : Access Control - u_book ☆ Application scope: Global Update set: Default [Global]

Type: record Operation: **read** Decision Type: Allow If

Name: u_book Description: Default access control on u_book

Applies To: No. of records matching the condition: 8 (empty)

Conditions

8. Submit.
9. Again, click New.
10. Select Table = Book [u_book].
11. Choose Operation = write.
12. Set Requires role = librarian.

servicenow All Favorites History Workspaces : Access Control - u_book ☆ Application scope: Global Update set: Default [Global]

Type: record Operation: **write** Decision Type: Allow If

Name: u_book Description: Default access control on u_book

Applies To: (empty)

Conditions

13. Submit.
14. Again, click New.
15. Select Table = Book [u_book].
16. Choose Operation = delete.

Access Control
u_book

Type record

Operation delete

Decision Type Allow if

Admin overrides

Protection policy -- None --

Name u_book

Description Default access control on u_book

Applies To No. of records matching the condition: 8
(empty)

Conditions

Access Control Rules have two decision types, and these types will behave differently depending on conditions.

1. Allow Access: Allows access to a resource if all conditions are met. Additional ACLs may grant access to records where this ACL has not granted access.
2. Deny Access: Denies access to a resource unless all conditions are met. Additional ACLs may not grant access to records where this ACL has denied access.

17. Set Requires role = librarian.

18. Submit.

Result: Students can only read books, librarians can read, write, and delete.

Access Control for Borrow Request Table

1. Open ServiceNow.
2. Click on All → search for Access Control.
3. Select System Security → Access Control (ACL).
4. Click New.
5. Select Table = Borrow Request [u_borrow_request].
6. Choose Operation = create.
7. Set Requires role = student.
8. Submit.
9. Again, click New.
10. Select Table = Borrow Request [u_borrow_request].
11. Choose Operation = read.
12. Set Requires role = student, librarian.

servicenow All Favorites History Workspaces : Access Control - u_borrow_request ☆ Application scope: Global
Update set: Default [Global]

Access Control
u_borrow_request

Conditions

Access Control Rules have two decision types, and these types will behave differently depending on conditions.

1. Allow Access: Allows access to a resource if all conditions are met. Additional ACLs may grant access to records where this ACL has not granted access.
2. Deny Access: Denies access to a resource unless all conditions are met. Additional ACLs may not grant access to records where this ACL has denied access.

More Info

Requires role

Role

- student
- u_borrow_request_user
- librarian

Security Attribute Condition

Local or Existing Local

13. Submit.
14. Again, click New.
15. Select Table = Borrow Request [u_borrow_request].
16. Choose Operation = write.
17. Set Requires role = librarian.
18. Submit.
19. Again, click New.
20. Select Table = Borrow Request [u_borrow_request].
21. Choose Operation = delete.
22. Set Requires role = librarian.
23. Submit

Result: Students can create and read their borrow requests, librarians can read, update, and delete requests.

Steps to Create Report:

1. Open ServiceNow.
2. Click on All → search for Reports.
3. Select Reports → View / Run.
4. Click on New.
5. Enter Name = Most Borrowed Books.
6. Select Source Type = Table.
7. Select Table = Borrow Request [u_borrow_request].
8. Choose Type = Bar Chart (or Pie Chart as needed).
9. In the Group by field, select Book.
10. In the Aggregate field, select Count.
11. Add a filter: Status = Approved (to count only issued books).
12. Click Run to preview the report.
13. Adjust chart options (Title, Labels, Legend) as required.
14. Click Save.
15. To publish, click Sharing → Add roles (student, librarian).

Phase 5: Deployment, Documentation & Final Presentation

Troubleshooting

Flow Execution & Logs

Verified end-to-end execution of the Borrow Request Approval Flow using Flow Designer Execution Details. Checked trigger conditions, approvals, and book issuance updates.

Reference Qualifier Issues

Debugged cases where already-issued books were still showing in the Borrow Request form. Corrected reference qualifier conditions to filter only available books.

UI Policy Behavior

Tested Borrow Request form to ensure Return Date became mandatory when Status = Issued. Verified field visibility and mandatory settings across portal and platform views.

Access Control Validation

Checked that students could only create/read their own Borrow Requests while librarians had full access. Adjusted ACL conditions to restrict unauthorized access.

Report Accuracy

Validated the “Most Borrowed Books” report to ensure only approved/issued requests were counted. Fixed grouping to display correct book titles.

Adherence to Timelines

Executed project based on the defined milestones:

- Creation of Roles
- Creation of Tables
- Adding Fields (Book & Borrow Request)
- Reference Qualifier Implementation
- Flow Designer Setup (Approval & Issuance)
- UI Policy Configuration
- Access Control Setup
- Report Creation (“Most Borrowed Books”)
- Testing & Validation

Innovation

End-to-End Automation: Automated the book borrowing process by integrating custom tables, Flow Designer, and approvals, eliminating manual tracking of borrow requests.

Optimized Request Handling: Implemented reference qualifiers to ensure only available books can be requested, reducing errors and duplicate borrow entries.

Maintainability & Scalability: Designed the solution using out-of-the-box ServiceNow features like Flow Designer, UI Policies, and ACLs, ensuring easy maintenance and future scalability.

User Experience Enhancements: Applied UI policies to make Return Date mandatory only when a book is issued, keeping forms clean and user-friendly for both students and librarians.

Document Technical Blueprint

Variables:

- Book Title and Author
- Borrower (auto-populated with current user, editable if borrowing for others)
- Borrow Date (auto-filled with current date)
- Return Date (mandatory only when book is issued)

Approval Use Cases:

- Librarian Approval required automatically when a student raises a borrow request.
- Auto-approval for requests made by librarian role (for testing or internal usage).

Fulfillment Task Use Cases:

- Catalog tasks created for the Librarian group to issue the book.
- Automatic update of Borrow Request status to “Issued” once approved.
- Automatic update of Borrow Request status to “Returned” when the book is marked as returned.

Document Setup Manual

· Provided step-by-step manual for recreating the solution in a new Personal Developer Instance (PDI):

- Role creation (Librarian, Student)
- Table creation (Book, Borrow Request)
- Field configuration for each table
- Reference qualifier setup for available books
- Flow Designer configuration (Approval → Issue → Return)

- UI Policy setup for mandatory Return Date
- Access Control rules for students and librarians
- Report creation (“Most Borrowed Books”)
 - Test data submission performed to validate borrow/return flow
 - Screenshots and order of operations included for clarity

Project Demo Video Planning

- Demo includes:
- Portal walkthrough: Home → Library Services → Submit Borrow Request
- Live book borrow request submission by a student
- Dynamic approval flow handled by librarian
- Email notification triggers for request approval and issuance
- Request and book status tracking (Requested → Approved → Issued → Returned)
- Includes narration for each step and role-based perspectives (Student, Librarian)

Visual Demonstration

Captured:

- o Form UI behaviour (e.g., Return Date becomes mandatory only when a book is issued)
- o Approval records in the Borrow Request timeline (Student request → Librarian approval → Issued)
- o Flow logs showing the end-to-end execution path (Request submitted → Approved → Issued → Returned)
- o Annotated visuals with labels and highlights to explain each stage clearly

Explanation Clarity

- Documentation and presentation follow logical, sequential structure:
- What → Why → How → Outcome
- Each step includes:
- Clear headings for every configuration (Roles, Tables, Flow, UI Policy, etc.)
- Labelled screenshots of ServiceNow setup and execution
- Explanatory notes for easy replication in a new PDI

Realism & Quality

- Workflow mimics real-world library operations, including:
 - Role-based approvals (Student requests → Librarian approves)
 - Data validations (book availability check, mandatory return date)
- Solution tested with realistic use cases and sample data (multiple students requesting the same book, overdue returns).

Scalability & Future Plan

Future enhancements considered:

- Auto-notifications for overdue books and reminders via email/SMS
- Integration with external library systems or digital catalog APIs
- Enhanced reporting dashboards (borrow frequency, student activity, overdue analysis)
- Expansion to other academic services (lab equipment requests, study room booking, digital resource access)

Conclusion

The Library Management Project successfully streamlined the process of borrowing and managing books within ServiceNow by integrating role-based access, automated approvals, and dynamic workflows. Students can easily submit borrow requests, while librarians efficiently review, approve, and issue books with proper validations. The solution ensures transparency through status tracking, notifications, and reporting features. By combining automation with usability, the project delivers a scalable foundation that can be extended to broader academic services in the future.