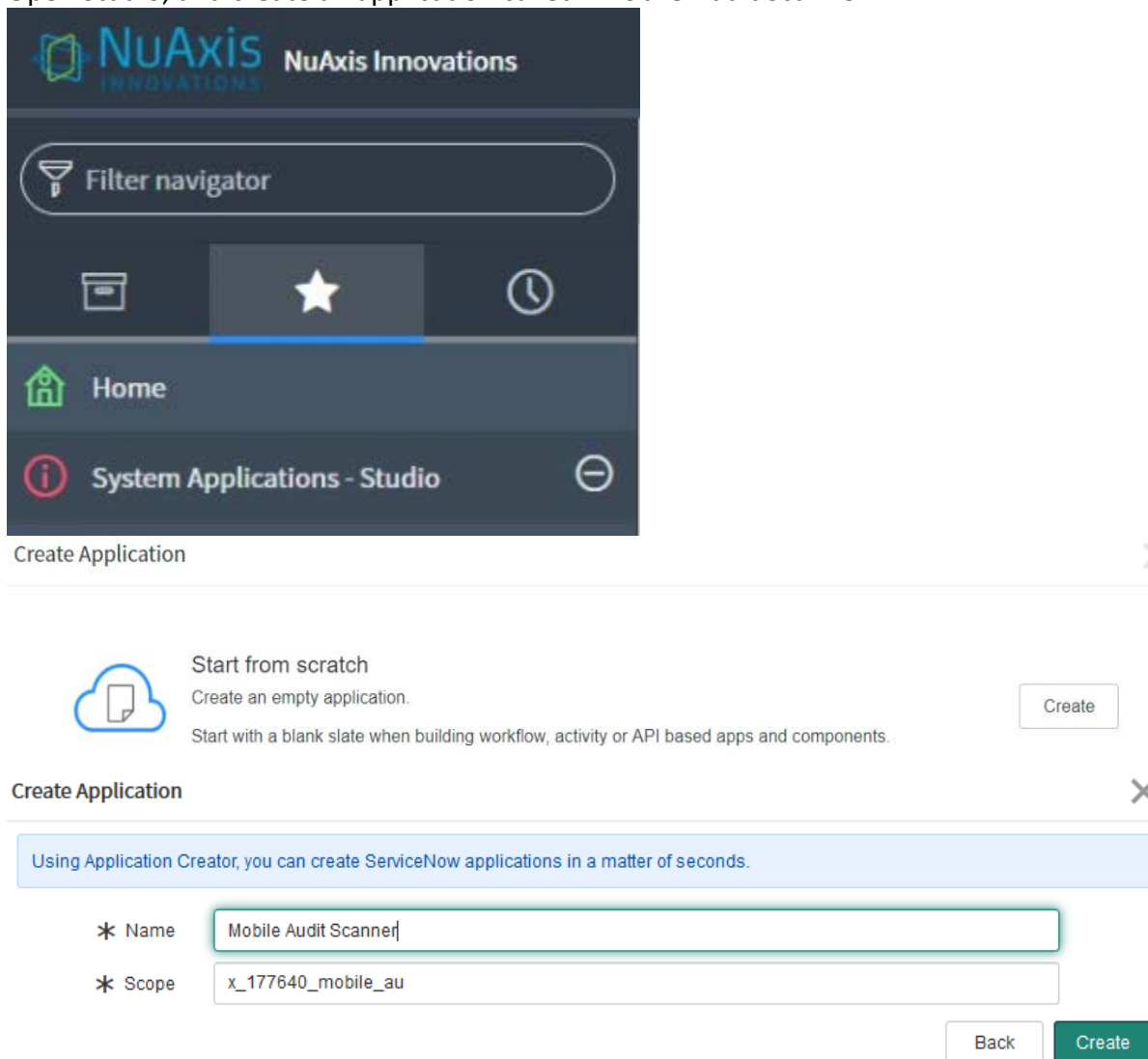


This application is from the ServiceNow Knowledge19 CreatorCon Session CCB0678: Mobile inventory scanner: Building a no-code phone application

Developing a mobile phone application for your company's yearly physical inventory can turn a nightmare of spreadsheets into a dream. Enter the mobile inventory scanner. In this session, you'll discover how — using the power of Madrid's new Mobile Studio and Flow Designer — any level of admin or developer can build a barcode-scanning and GPS-tagging powerhouse of automation.

Below you will find the steps (with screenshots) required to build the framework of this scoped application. If wish to see every step in more detail please watch the video linked at the end of the document. But I recommend you use this document as a framework to build the application you need instead of cloning mine.

1. Open Studio, and create an application called “Mobile Audit Scanner”.



Create Application

Start from scratch  
Create an empty application.  
Start with a blank slate when building workflow, activity or API based apps and components.

Create

Create Application

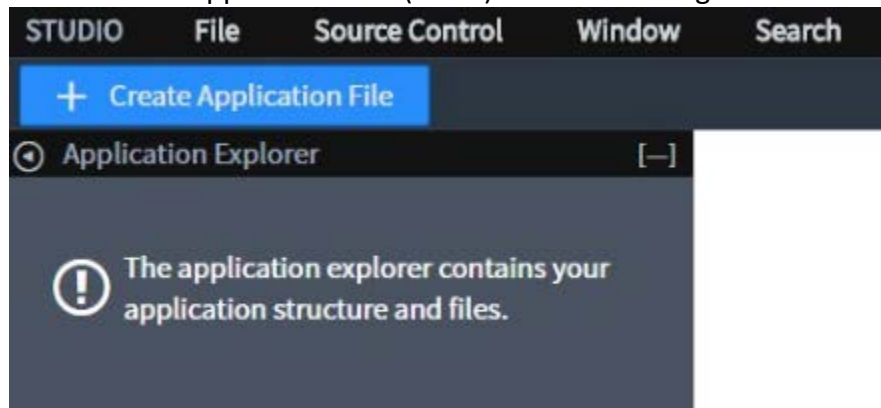
Using Application Creator, you can create ServiceNow applications in a matter of seconds.

\* Name Mobile Audit Scanner

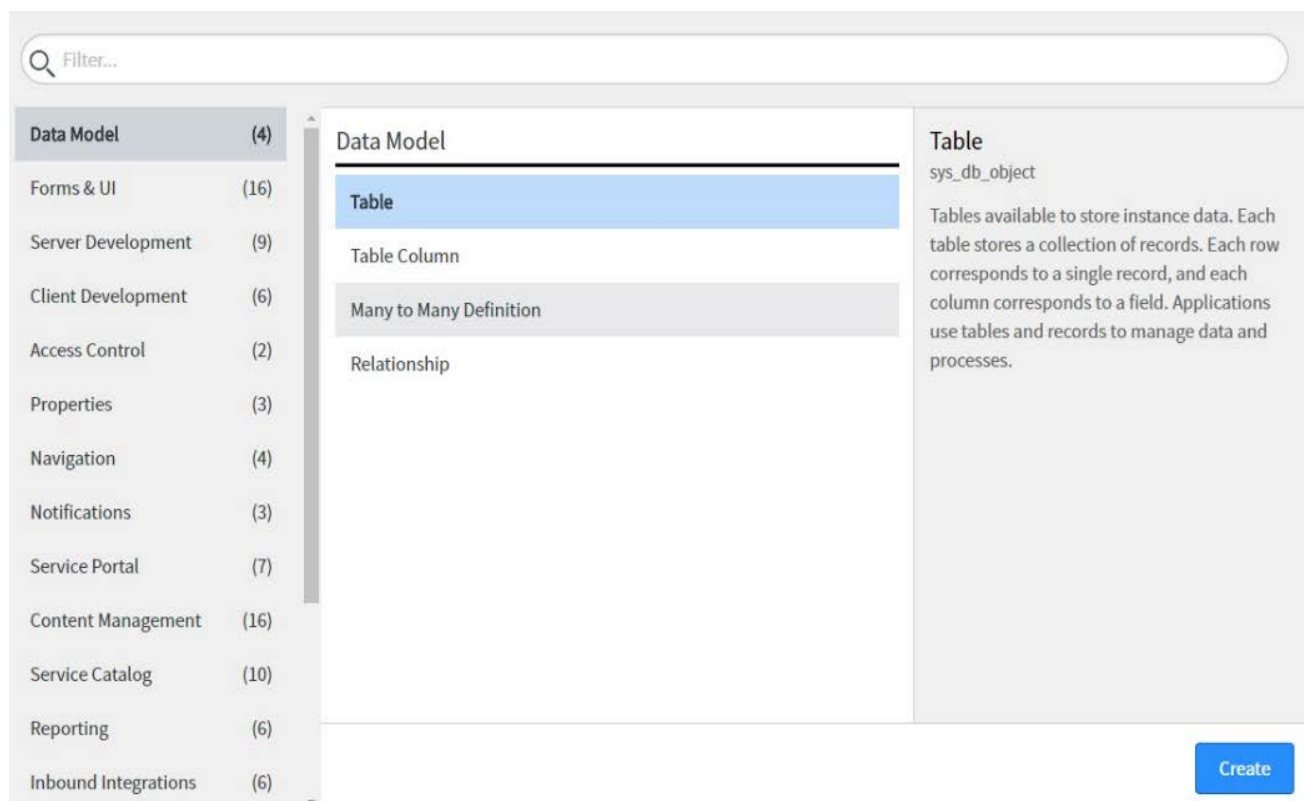
\* Scope x\_177640\_mobile\_au

Back Create

2. Create a new Application File. (Table) called "Scan Log".



Create Application File




3. While creating the table.
  - a. Uncheck the Create Module Box
  - b. Check all boxes under the Application Access Tab
  - c. Check the Auto Number Box under the Controls Tab
  - d. Use the Prefix of LOG for the Auto Number Field
  - e. Change the role under the checked box for create ACL's to ITIL


Table  
New record




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](#)

\* Label  

\* Name


Extends table  

Application  

Create module ☐



Columns Controls Application Access

Accessible from  

Can read ☒

Can create ☒

Can update ☒

Can delete ☒

Allow access to this table via web services ☒

Allow configuration ☐

Columns Controls Application Access

Extensible ☐

Live feed ☐

Use auto-numbering to define a sequential identifying code made up of a prefix, a base number and a padding value to ensure a consistent format

Auto-number ☒



Prefix

Number

Number of digits

Security Rules (ACLs) are required if anyone other than an administrator needs to work with this table. Creating default security rules will grant full access to this table to anyone with the user role you specify.

\* Create access controls ☒

\* User role   

4. Create 8 new fields on this table. (Found under the Columns Tab) (Number will already be there).

Table Columns

New

Search

for text

▼

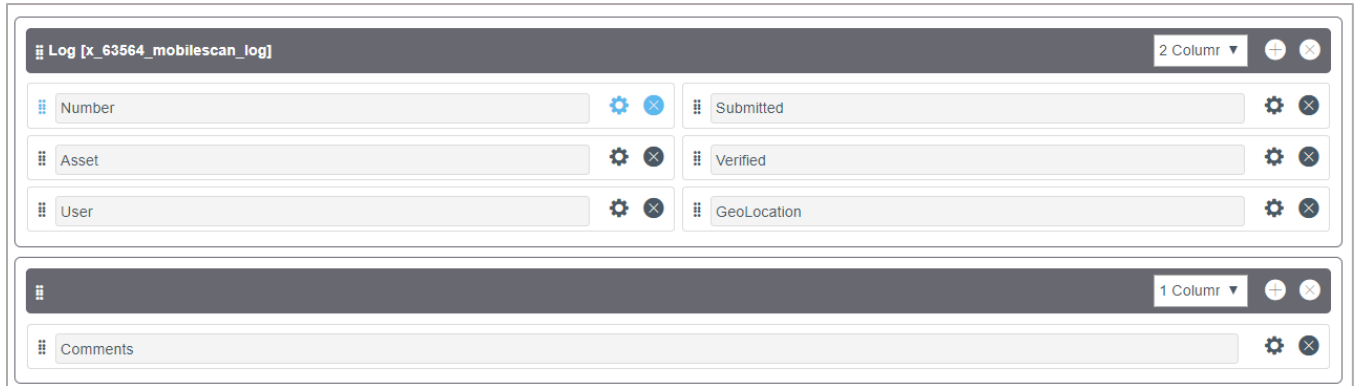
Search

Dictionary Entries

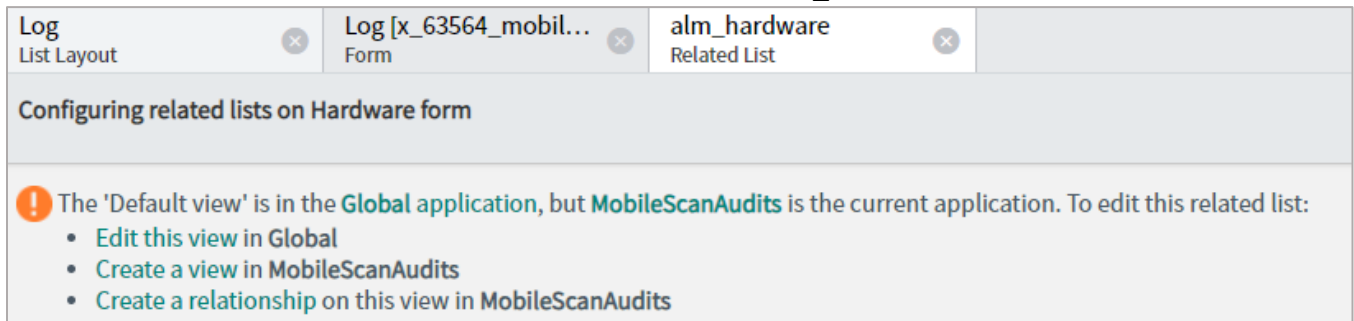
		Column label	Type	Reference	Max length
		Asset	Reference	Hardware	32
✗		Barcode	String	(empty)	40
✗		Comments	String	(empty)	4,000
✗		GeoLocation	String	(empty)	255
✗		Location	Reference	Location	32
✗		Number	String	(empty)	40
✗		Submitted	True/False	(empty)	40
✗		User	Reference	User	32
✗		Verified	True/False	(empty)	40

- Asset** (Reference to the **alm\_hardware** table)
- Barcode** (String 40 Characters Long)
- Comments** (String 4000 Characters Long)
- GeoLocation** (String 255 Characters Long)
- Location** (Reference to the **cmn\_location** table)
- Submitted** (True/False)
- User** (Reference to the **sys\_user** table)
- Verified** (True/False)

5. Create a cleaner form layout for the Scan Log Table.

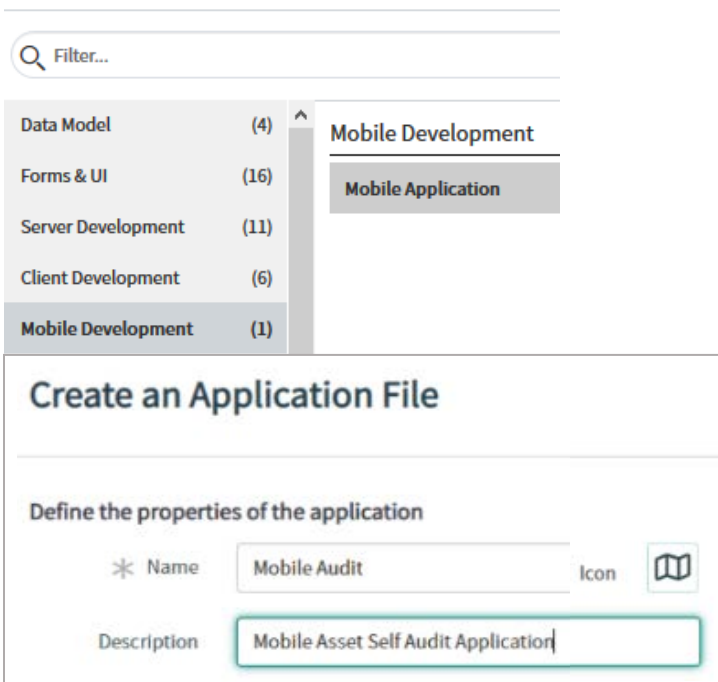


6. Create a Related List in a new View “Mobile Audit” for the alm\_hardware form.



7. Create a Mobile Application in Studio.

Create Application File



## 8. Create Folders/Sections for Applets.

Mobile Audit

Application

Properties

Active ☒

Save

Delete

Applets List

Create a folder

Self Audit

+ Create an Applet

Compliance Verification

+ Create an Applet

## 9. Create an Applet “My Asset Audits Due” List Type (Repeat this step for all new applets needed).

Create an Applet

×

Define the properties of the applet

\* Name

My Asset Audits Due

\* Icon

Description

Description

Choose the screen(s) template

Screen(s) Preview

List

Map

Calendar

Employee Directory

Grouped List

URL

You can select or deselect the screen(s) for this template.

List

Details

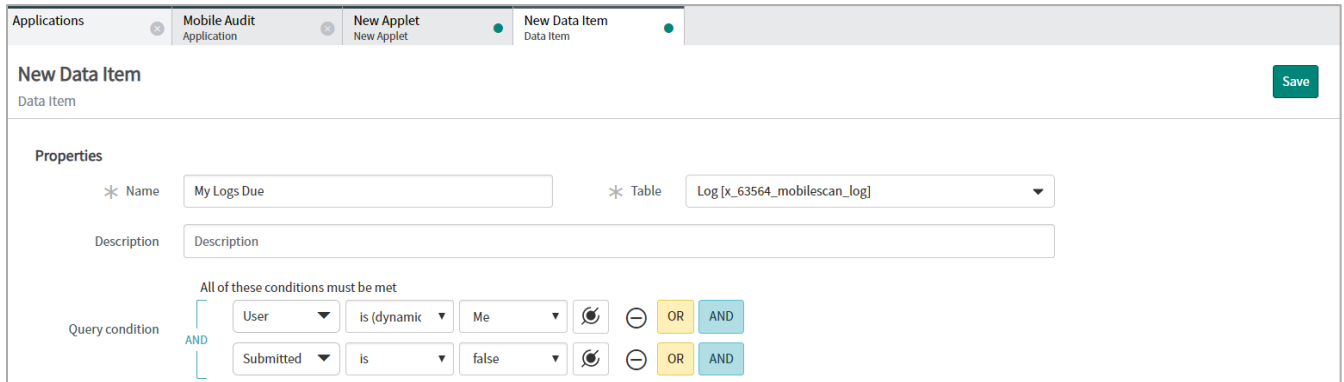
Activity Stream

Related List

Cancel

New

10. Create a Data Item – Clicking the + Sign inside of the applet.  
Set the Query for the new data item. (User is Dynamic Me & Submitted is False).



**New Data Item**  
Data Item

**Properties**

\* Name: My Logs Due \* Table: Log [x\_63564\_mobilesan\_log]

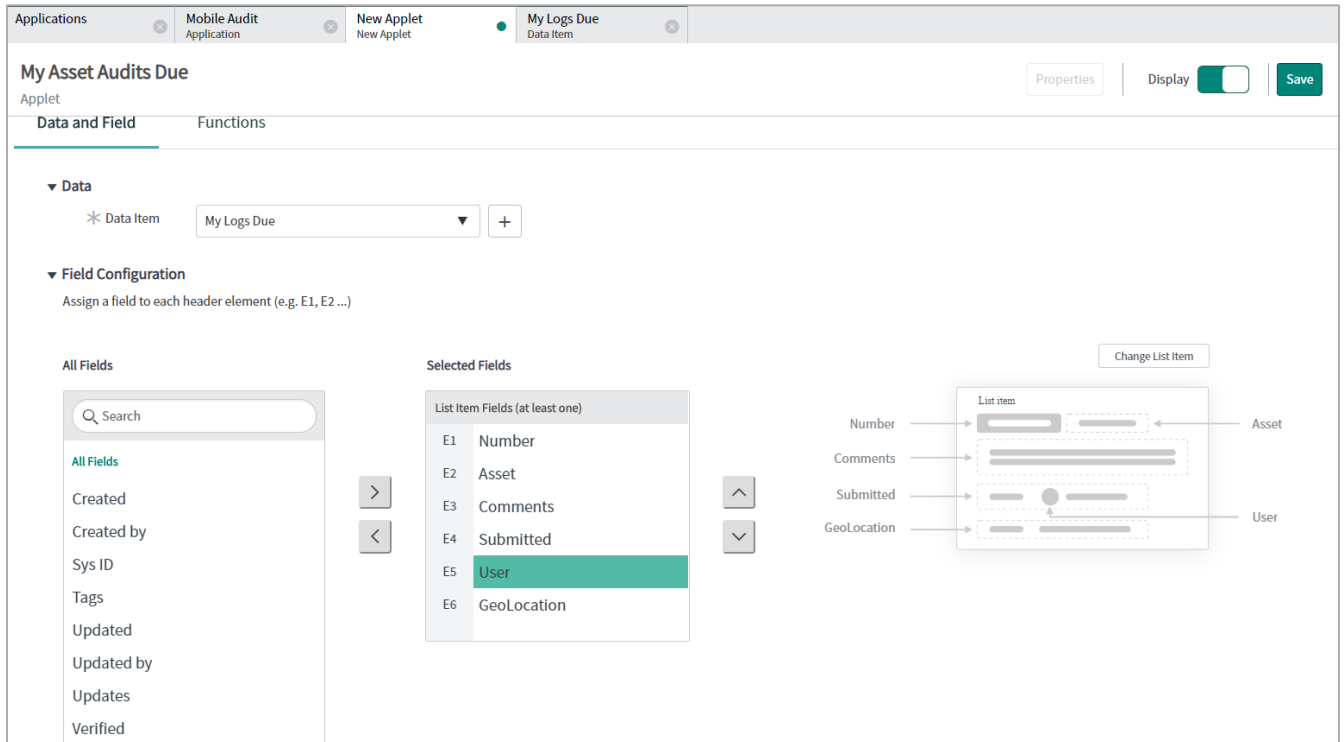
Description: Description

**Query condition**  
All of these conditions must be met

AND

User is dynamic Me Submitted is false

11. Configure the Data View Screens. (Data and Field & Functions Tabs).  
Use the Duplicate Fields button on the Function and Data Screen Tabs.  
You can dot-walk to additional fields, move dot-walked fields first.



**My Asset Audits Due**  
Applet

**Data and Field** Functions

▼ Data  
\* Data Item: My Logs Due

▼ Field Configuration  
Assign a field to each header element (e.g. E1, E2 ...)

**All Fields**

Search

All Fields

Created  
Created by  
Sys ID  
Tags  
Updated  
Updated by  
Updates  
Verified

**Selected Fields**

List Item Fields (at least one)

E1 Number  
E2 Asset  
E3 Comments  
E4 Submitted  
E5 User  
E6 GeoLocation

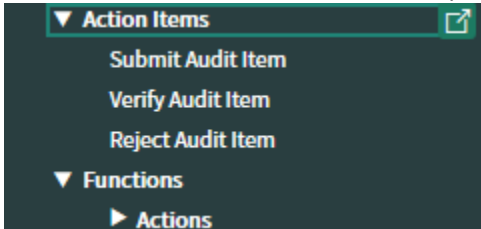
**Change List Item**

Number  
Comments  
Submitted  
GeoLocation

List item

Asset  
User

12. Create Function and Action Items (Within Studio you will need to hover over the menu to pop-out).



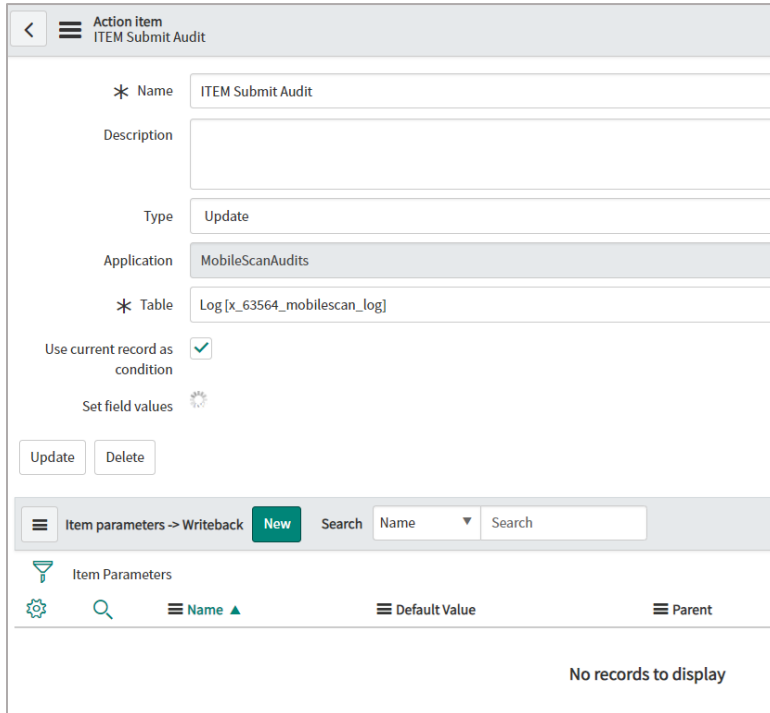
▼ Action Items

Submit Audit Item  
Verify Audit Item  
Reject Audit Item

▼ Functions

► Actions

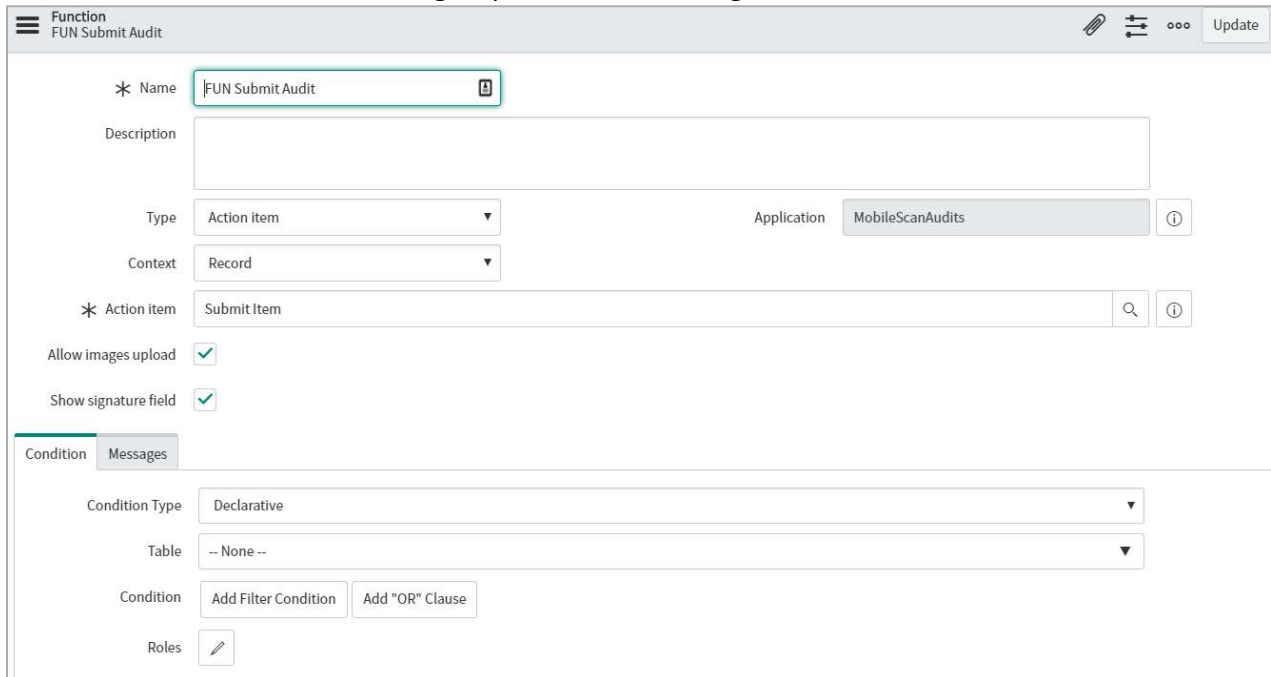
13. Create the Action Item First, Just set the (Use current record as condition, and select the Log Table.



The screenshot shows the 'Action item' configuration form for 'ITEM Submit Audit'. The form includes the following fields and options:

- Name:** ITEM Submit Audit
- Description:** (Empty text area)
- Type:** Update
- Application:** MobileScanAudits
- Table:** Log [x\_63564\_mobilesan\_log]
- Use current record as condition:** ☒
- Set field values:** (Gear icon)
- Buttons:** Update, Delete
- Item parameters -> Writeback:** New, Search, Name (dropdown), Search (input)
- Item Parameters:** (Filter icon, Search icon, Name (dropdown), Default Value, Parent)
- Footer:** No records to display

14. Create the Function Second, you will need to tie it to the Action Item just created. Name it and select the Allow Image Upload and Show Signature Checkboxes.



The screenshot shows the 'Function' configuration form for 'FUN Submit Audit'. The form includes the following fields and options:

- Name:** FUN Submit Audit
- Description:** (Empty text area)
- Type:** Action item
- Application:** MobileScanAudits
- Context:** Record
- Action item:** Submit Item
- Allow images upload:** ☒
- Show signature field:** ☒
- Condition:**
  - Condition Type:** Declarative
  - Table:** -- None --
  - Condition:** Add Filter Condition, Add "OR" Clause
  - Roles:** (Edit icon)
- Messages:** (Tab)
- Buttons:** Update



# 15. Create a UI Parameters for Automatic and User Input. (On the Function)

Get Comments – Manual Input (String the user enters)

Get GPS – Auto Fill (Mobile device GPS signal)

Get Barcode – Manual Input (Mobile Camera Scanner)

UI parameters (3)

Action parameters mappings (3)

☰

UI parameters

New

Search

for text

▼

Search

◀◀

▶▶

1

🔍

Button = Submit Audit Function

<div>⚙️</div>	<div>🔍</div>	<div>☰ Name</div>	<div>☰ Button parent table</div>	<div>☰ Carried</div>	<div>☰ Constant value</div>	<div>☰ Default value</div>	<div>☰ Default value type</div>	<div>☰ Field name</div>	<div>☰ Input source</div>
<div><input type="checkbox"/></div>	<div>📘</div>	<div>Get Comments</div>		<div>false</div>			<div>None</div>		<div>User input</div>
<div><input type="checkbox"/></div>	<div>📘</div>	<div>Get GPS</div>		<div>false</div>			<div>None</div>		<div>Auto fill</div>
<div><input type="checkbox"/></div>	<div>📘</div>	<div>Get Barcode</div>		<div>false</div>			<div>None</div>		<div>User input</div>

# 16. Create an Item Parameter per desired write-back (On Item) (Repeat per UI Parameter created).

Item Parameter

New record

\* Name

get comments

Application

MobileScanAudits

\* Parent table

Action item [sys\_sg\_write\_back\_action\_i...

Default Value

\* Parent

Action item: ITEM Submit Audit

Type

String

Submit

# 17. Create Action Parameter (Linkage between Item and Functions) (Repeat for each Item Parameter) .

Action parameters mapping

get comments [Default view]

\* Button

FUN Submit Audit

\* Item parameter

get comments

Application

MobileScanAudits

\* UI parameter

Comments

Update

Delete

# 18. On the Action Item, now that the linkage parameter are there you can fill in what (set field values).

\* Table

Scan Log [x\_177640\_mobile\_scan\_log]

Use current record as condition

☒

Set field values

Comments

get comments2

Barcode

get barcode2

GeoLocation

get gps2

19. Now that the Actions and Functions are completely build Navigate back to the Applet screen.  
Set the function where you want it by clicking the + Icon.

### My Audits Due

Applet

Properties

My Audits Due (primary screen)

Details

#### Screen Configurations

Data and Field

Functions

##### Top Menu Functions

Please add functions of "Global" context into top menu.



Label	Function	Function Type	
-------	----------	---------------	--

No Top Menu Functions

##### Swipe Functions

Please add functions of "Global" or "Record" context into swipe.

Label	Swipe Direction	Function	Function Type	
-------	-----------------	----------	---------------	--

	Submit Audit	Left	Submit Audit Function	Action	
---	--------------	------	-----------------------	--------	---

20. Create a FLOW in Flow Designer to populate the LOG Records Yearly - Titled (Populate Audit Logs).  
Create Application File

Filter Results (1)

Data Model (4)

Filter Results

Flow Flow Designer

Title the Flow, and set a Trigger Condition (Repeat every 365 Days shown below).

servicenow Flow Designer

Populate Audit ...

FLOW Populate Audit Logs (Yearly)

TRIGGER

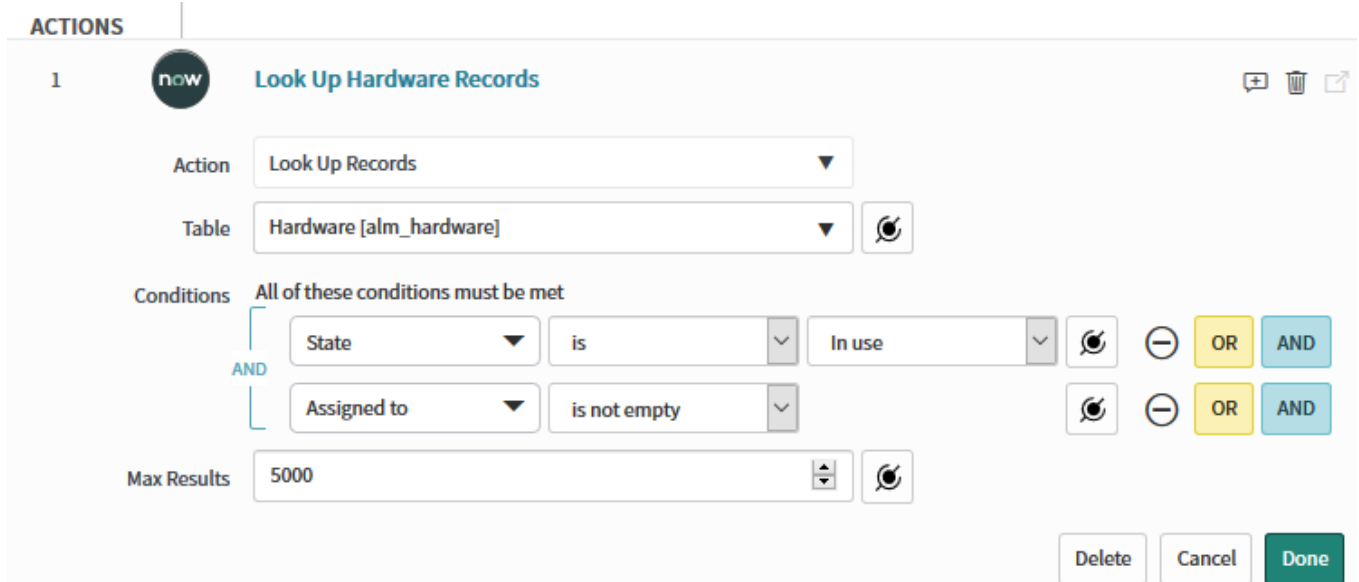
now Repeat every 365 day(s) 00 hour(s) 00 minute(s) 01 second(s)

Trigger Repeat

Repeat 365 Days

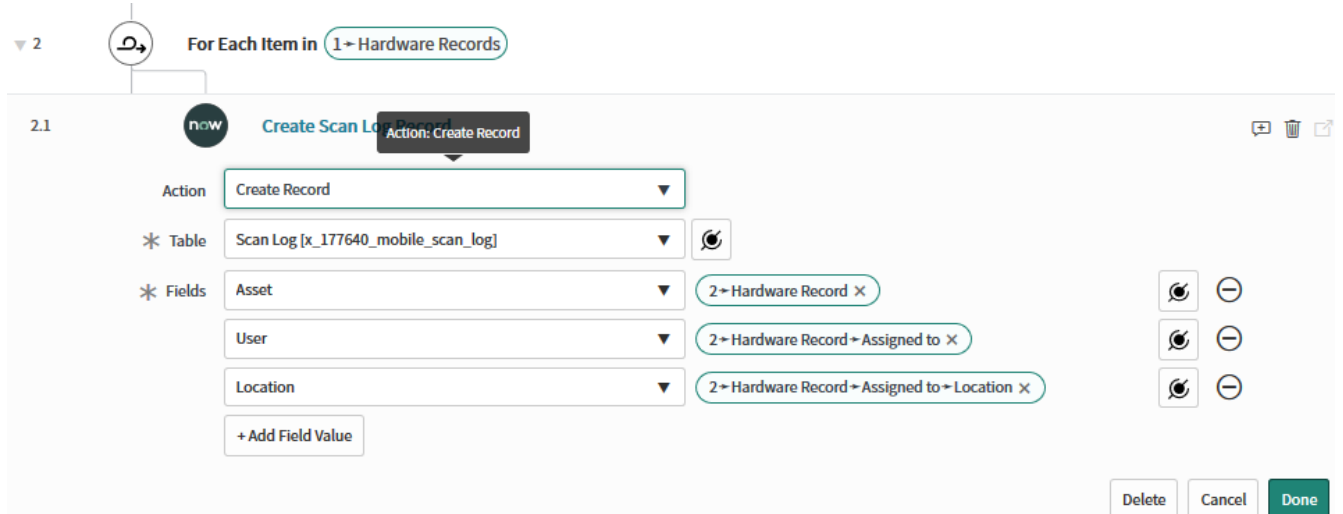
0 h 0 m 1 s

21. Setup an Action – That gathers all records on the “alm hardware” table that are In-Use and the assigned to (is not empty), also you will want to increase the Max Results to be appropriate.



The screenshot shows the configuration for the 'Look Up Hardware Records' action. The 'Action' dropdown is set to 'Look Up Records'. The 'Table' dropdown is set to 'Hardware [alm\_hardware]'. The 'Conditions' section is set to 'All of these conditions must be met' and contains two conditions: 'State is In use' and 'Assigned to is not empty', connected by an 'AND' operator. The 'Max Results' field is set to '5000'. The interface includes 'Delete', 'Cancel', and 'Done' buttons at the bottom right.

22. Then you will build a “For Each Loop” to create a new record on the “Scan Log”. You will set the following fields on that table shown below. (Asset, User, & Location)



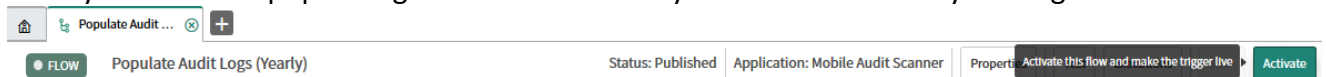
The screenshot shows the configuration for the 'Create Scan Log' action, which is part of a 'For Each Item in' loop. The 'Action' dropdown is set to 'Create Record'. The 'Table' dropdown is set to 'Scan Log [x\_177640\_mobile\_scan\_log]'. The 'Fields' section lists three fields: 'Asset', 'User', and 'Location'. Each field has a corresponding value field: '2 → Hardware Record', '2 → Hardware Record → Assigned to', and '2 → Hardware Record → Assigned to → Location'. The interface includes 'Delete', 'Cancel', and 'Done' buttons at the bottom right.

23. To populate the table while testing the application click “Test”.



The screenshot shows the application interface for 'Populate Audit Logs (Yearly)'. The 'Test' button is highlighted, and a tooltip indicates 'Simulate a trigger and execute the flow'. The interface also shows the 'Status: Published' and 'Application: Mobile Audit Scanner'.

24. Once you know it is populating the records correctly activate the Flow. By clicking the Activate Button.



The screenshot shows the application interface for 'Populate Audit Logs (Yearly)'. The 'Activate' button is highlighted, and a tooltip indicates 'Activate this flow and make the trigger live'. The interface also shows the 'Status: Published' and 'Application: Mobile Audit Scanner'.

You have now seen all the steps to build out the framework of this application. You will need to repeat steps 09-18 to build out additional applets to see other lists and maps, and have functions and actions performed in those applets. This walkthrough was to show you how to build the application, knowing you will want to build it to your company's needs.

If you care to see the complete step by step build I have recorded a video of it in its entirety. Or you can install the application though GitHub to your PDI, or pre-production instance. But I do recommend you tailor this to your organization instead of using it straight from my repository.

Video Link: <https://youtu.be/6VXZKtkpNGg>

Git-Hub Link: <https://github.com/NuAxis/ServiceNow-Mobile-Audit-Scanner> (Repository to Fork)

Please contact [casey.barela@nuaxis.com](mailto:casey.barela@nuaxis.com) with any questions.