## LD1: Platform Overview & Navigation (7%)

- [ ] ServiceNow Platform Overview
  - [ ] Modernized Work Experience
  - [ ] Useful Resources
- [ ] Platform Capabilities & Services
  - [ ] ITSM Incident Management
  - [ ] Service Operations Workspace
  - [ ] Employee Center Portal
- [ ] The ServiceNow Instance
- [ ] Next Experience Unified Navigation
  - [ ] All Menu
  - [ ] Favorites / Pins
  - [ ] History
  - [ ] Workspaces
  - [ ] Elevation
  - [ ] Impersonation

## LD2: Instance Configuration (11%)

- [ ] Personalizing / customizing the instance
  - [ ] Configuration
- [ ] Installing applications and plugins
- [ ] Users/Groups/Roles *
- [ ] Common user interfaces in the Platform

## LD4: Self Service & Automation (20%)

- [ ] Knowledge Management
  - [ ] Architecture
  - [ ] User Criteria Workflows
- [ ] Service Catalog
- [ ] Flow Designer
- [ ] Virtual Agent

## LD5: Database Management (27%)

- [ ] Data Schema
  - [ ] Tables, Records and Fields
  - [ ] Relationships
  - [ ] Schema Map
- [ ] Application / Access Control
- [ ] Importing Data
- [ ] CMDB and CSDM:
  - [ ] CMDB
  - [ ] CSDM
  - [ ] Discovery / Service Mapping
  - [ ] Dependencies / View

* Not specifically listed within *Exam Scope* but covered within *SNAF* course material and bonus labs.

## LD3: Config Apps for Collaboration (20%)

*MD3: Configure Applications for Business*

- [ ] Lists, Filters and Tags
  - [ ] Filters / Context Menus / Tags
- [ ] List and Form Anatomy
  - [ ] Lists Views
  - [ ] Workspaces
  - [ ] Forms
- [ ] Form Configuration
  - [ ] Form Design / Form Layout
- [ ] Advanced Form Configuration
  - [ ] Choice / Reference Lists

*MD5: Enable Productivity*

- [ ] Reporting, Dashboard, Performance Analytics
  - [ ] Reporting
  - [ ] Dashboards
  - [ ] Performance Analytics
  - [ ] Tags
- [ ] Form templates and saving options
  - [ ] Form Templates
- [ ] Task Management
- [ ] Visual Task Boards (VTBs)
- [ ] Quick Messages *
- [ ] Notifications
- [ ] Sidebar *
- [ ] Integration Hub *
- [ ] Automation Center *
- [ ] Predictive Intelligence *

*MD 6: Manage an Instance*

- [ ] Performance Dashboard *
- [ ] System Diagnostics *
- [ ] Instance Scan *
  - [ ] Checks / Suites / Findings / Cleanup
- [ ] Health Scan *
- [ ] IT Adoption Journeys *
- [ ] Upgrades and Releases *

## LD6: Data Migration and Integration (15%)

- [ ] Client Scripts *
- [ ] UI Policies
- [ ] Business Rules
- [ ] System Update Sets
- [ ] ATF / App Engine Studio *

contains Practice Tasks

## LD1: Platform Overview & Navigation (7%)

- ➔ ServiceNow Platform Overview
  - ◆ **Modernized Work Experience**
    - 💡**Knowledge** [*Welcome to ServiceNow*]:
      - ☐ Know both the founder and the CEO of ServiceNow
        - ◆ Founder: Bill Luddy
        - ◆ CEO: Bill McDermott
      - ☐ Understand the capabilities of the platform
        - (single system of record, single data model, configurable, plug 'n play, etc.)
      - ☐ Know the 4 ServiceNow workflows
        - ◆ IT Workflows, Customer Workflows, Creator Workflows, Employee Workflows
      - ☐ Know the 4 ServiceNow interfaces
        - ◆ The Next Experience (Unified Navigation)
        - ◆ Now Mobile App
        - ◆ Service Portal
        - ◆ Employee Center
  - ◆ **Useful Resources**
    - 💡**Knowledge** [*Welcome to ServiceNow*]:
      - ☐ Know the 4 useful resources identified within this module:
        - ◆ Now Learning, ServiceNow Community, Service Global Events, ServiceNow Developer Site
- ➔ Platform Capabilities & Services
  - ◆ **ITSM Incident Management**
    - 💡**Knowledge** [*SNAF Module 1 - ServiceNow - The Modernized Work Experience*]:
      - ☐ How to navigate to an Incident from the **All** menu
        - ◆ **All > Incident > Create New**
      - ☐ Understand Incident Lifecycle (*State*s from New -> Closed)
        - ◆ New, In Progress, On Hold, Resolved, Closed, Canceled
      - ☐ Familiar with Incident handling within Classic View
        - ◆ Assigning fields, updating notes, viewing activity steam, etc.
      - ☐ Understand Incident Assignment / Groups
        - ◆ Assignment Group may be defined by the Service, or assigned manually
      - ☐ Know difference between *Work Notes* / *Additional Comments*
        - ◆ Work notes are viewable only within the Task
        - ◆ Additional Comments are viewable by the requestor
      - ☐ Incident prioritization and handling based on *Impact* & *Urgency* fields
        - ◆ Setting Impact and Urgency to 1 automatically sets Priority to 1
      - ☐ Base understanding of Incident SLAs and why they exist
        - ◆ Provides expected timing for ticket resolution based on Priority
      - ☐ *Related Lists* section
        - ◆ Based on how Related Lists are configured, this section shows other database elements that pertain to the current record (ex. SLAs, CIs, Child Incidents, etc.)
    - 📝**Practice Tasks** [*SNAF Lab 1.2 - Create an Incident using classic lists and forms*]
      - ☐ Impersonate **Beth Anglin**
      - ☐ Create a new Incident
      - ☐ Assign an incident to individual and/or a group (or auto-assign)
      - ☐ Add Notes / Comments
      - ☐ Navigate to and access an in-progress incident through *My Group's Work*
  - ◆ **Service Operations Workspace**
    - 💡**Knowledge** [*SNAF Module 1 - ServiceNow - The Modernized Work Experience*]:
      - ☐ How to navigate to the workspace
        - ◆ **Workspaces** in top header > click **Service Operations Workspace**
      - ☐ Widgets on the dashboard and what they do; how to navigate within the workspace
        - ◆ Incidents assigned, Incident SLAs, Unassigned incidents, Catalog tasks assigned (either To Me, or To My Team with top-right select box)

- Picking a top widget will determine what list of records to show on the bottom, clicking a record opens in new tab
- [ ] Understand how Classic View (**CV**) is mapped within Workspace (**WS**) view
  - **WS** *Overview*:
    - *Summary* -> **CV** *Description* and *Short Description*
    - *Impact* -> **CV** Impact
    - *Affected CIs* -> **CV** *Affected CIs* (under Related Lists)
  - ***WS** Details* -> All other **CV** Incident fields
  - ***WS** Related records* -> **CV** *Related Lists*
- [ ] *Compose* and *Activity* sections for adding/reviewing notes
  - Under *Details* tab to view **CV** *Activity Stream*, *Work Notes* and *Comments*
- [ ] The 5 right-panel support tools and what they do
  - Record Information, Agent Assist, Experts On-Call, Attachments, Templates
- [ ] The *Related Records* section
  - Same as **CV**, except **CV** tabs displayed as **WS** List on left, with **CV** tab content displayed in **WS** frame to right of list
- [ ] List view
  - Brings up a tree view with **Categories** and **UX Lists** on the left, content displays on right

📝**Practice Tasks** [*SNAF Lab 1.3 - Work an Incident in Workspace view*]

- [ ] As **Beth Anglin**, Open above incident in Workspace view
- [ ] Explore Workspace layout and Update both Work notes and Comments
- [ ] Resolve the Incident

- **Employee Center Portal**

💡**Knowledge** [*SNAF Module 1 - ServiceNow - The Modernized Work Experience*]:

- [ ] How to navigate to the portal from the **All** menu
  - **All > Self-Service > Employee Center**
- [ ] Basic Layout and functionality of the portal
  - (*My Tasks*, *My Requests*, *Favorites)*
  - Knowledge, Search, Virtual Agent, etc.

📝**Practice Tasks** [*SNAF Lab 1.4 - Check Incident status using Employee Center IT Portal*]

- [ ] Impersonate **Fred Luddy** to view the incident in Employee Center
- [ ] Update the existing request
- [ ] Impersonate **Beth Anglin** to view recent update from employee on incident
- [ ] Add more notes and resolve incident

➔ The ServiceNow Instance

💡**Knowledge** [*SNAF Module 2 - Explore the Power of the ServiceNow Platform*]:

- [ ] Understand the concept of **Baseline Implementation**
  - Base installation of ServiceNow environment prior to any config/customization
- [ ] Understand the difference between Configuration vs. Customization
  - **Config:** what can be done without code in the platform using built-in functionality
  - **Customization**: adding additional functionality through plug-ins, scripts, etc.
- [ ] Know the **3** environment types of instances:
  - Production, Non-Production, (Dev/QA/UAT/Staging/etc.)
- [ ] Understand concept of **single-tenant** architecture
  - This means that each instance features an **isolated database** containing data, applications and customizations

➔ Next Experience Unified Navigation

- **All Menu**

📝**Practice Tasks** [*SNAF Lab 1.2 - Create an Incident using classic lists and forms*]

- [ ] Familiarize with how to find applications/modules using the filter search
  - minimum 2 chars to start filtering results to applications and modules based on role visibility and access

- **Favorites / Pins**

📝**Practice Tasks** [*SNAF Lab 1.2 - Create an Incident using classic lists and forms*]

- [ ] How to add/remove/customize Favorites

        - Click the **star icon** on contextual app pill, or add from various control menus
    - [ ] How to Pin and UnPin to content frame
        - Click the tab in the header and then click the **pin icon** to pin menu to the left

- **History**
    - 📝**Practice Tasks** [*SNAF Lab 1.2 - Create an Incident using classic lists and forms*]
        - [ ] Be familiar with usage and understand which items will and will **not** appear here
            - Homepages, UI Pages
- **Workspaces**
    - 📝**Practice Tasks** [*SNAF Lab 1.3 - Work an Incident in Workspace view*]
        - [ ] Be familiar with accessing workspaces (ex. *Service Operations Workspace*)
            - **Workspaces** in top header > click **Service Operations Workspace**
- **Elevation**
    - 💡**Knowledge** [*SNAF Module 1 - ServiceNow - The Modernized Work Experience*]
        - [ ] How to perform elevation and how it is displayed
            - Open **User Menu** > **Elevate Role** - displays as red border with up arrow
        - [ ] What options exist to elevate to and why you would use them
            - security_admin: allows you to manage Access Controls
            - worlpace_admin: allows you to manage workspace lists and categories
- **Impersonation**
    - 💡**Knowledge** [*SNAF Module 1 - ServiceNow - The Modernized Work Experience*]
        - [ ] How to perform impersonation and how it is displayed
            - Open **User Menu** > **Impersonate user** - displays as red border with eye icon
        - [ ] Which role(s) are required to perform impersonation
            - **admin** or **impersonator** role
        - [ ] Know **3** recommended roles to create for testing impersonation
            - **admin**: have a test administrator setup separate from the main System Admin
            - **ess**:  Employee Self-Service (allows testing as **requester** role)
            - **itil**: IT Infrastructure Library (allows testing as **process user** role)
        - [ ] System property required to log impersonations in the System Log
            - **glide.sys.log_impersonation**
    - 📝**Practice Tasks** [*SNAF Lab 1.2 - Create an Incident using classic lists and forms*]
        - [ ] Impersonate another user

## LD2: Instance Configuration (11%)

- ➔ Personalizing/Customizing the instance
  - ◆ **Configuration**
    - 💡**Knowledge** [*SNAF Module 2.1*]
      - ☐ How to navigate to and tailor the Company logo
        - ◆ **All > System Properties > My Company**
      - ☐ How to navigate to and update the Company name
        - ◆ **All > User Administration > Companies**
      - ☐ How to navigate to and mark the instance with tab/context pill
        - ◆ **All > System Properties > System** and update *Browser tab title* to **DEV**
      - ☐ How activate and apply a different theme
        - ◆ **sys_properties.list** and update *glide.ui.polaris.dark_themes_enabled* to **TRUE**
    - 📝**Practice Tasks** [*SNAF Lab 2.1 - Configure an Instance*]
      - ☐ Navigate to **System Properties** and update Company Logo
      - ☐ Navigate to **User Administration/Organization** and update Company name
      - ☐ Navigate to **System Properties** and update *Browser tab title* property
      - ☐ Update system property to enable dark theme
- ➔ Installing Applications and Plugins
  - 💡**Knowledge** [*SNAF Module 2.2 - Install Applications*]
    - ☐ How to navigate to and access **System Applications**
      - ◆ **All > System Applications > All Available Applications > All**
    - ☐ How to navigate to and access Plugins via **System Definition**
      - ◆ **All > System Definition > Plugins**
    - ☐ Understand Plugins concepts (demo data, roles required, related plugins)
      - ◆ **docs.servicenow.com** to identify the Plugin required
        - Review *Product Details*
        - Review *View Dependencies and Licensing Requirements*
        - Demo data should **only** be installed in non-production instances
    - ☐ Understand the concept of **Application Scoping** - what does it protect?
      - ◆ restricts access to an application's files and data from other applications/services.
  - 📝**Practice Tasks** [*SNAF Lab 2.2 - 2.3 - Locate application information and install Plugin*]
    - ☐ Use docs.servicenow.com to find desired application information
    - ☐ Perform installation successfully, ensure demo data is loaded correctly
- ➔ Users/Groups/Roles
  - 💡**Knowledge** [*SNAF Module 2.4 - Install Applications*]
    - ☐ The **5** Persona types and what each privilege is used for
      - ◆ *System Administrator*: provides access to almost all roles except for **HR** and **Security Operations**
      - ◆ *Specialized Administrator*: a user who manages specific applications (ex. **HR**, **Knowledge Base**)
      - ◆ *Process User*: a user who has one or more roles and provides a specific function within the organization – typically would include **itil** and **approver_user** roles
      - ◆ *Approver*: user who has the ability to view/modify approvals but does not have any other roles
      - ◆ *Requester*: user who has no other roles but still needs to be able to submit and manage their own requests
    - ☐ What **User**s can do in the platform and the system table they have a record on
      - ◆ An individual who has been granted access to the instance (**sys_user** table)
    - ☐ What **Groups** can do in the platform and the system table they have a record on
      - ◆ Set of users that share a common purpose (**sys_user_group** table)
    - ☐ What **Roles** are used for in the platform and the system table they have a record on
      - ◆ Collection of permissions (**sys_user_role** table)
    - ☐ Where and how to modify **Users** / **Groups** / **Roles**
      - ◆ **All > User Administration > Users**
      - ◆ **All > User Administration > Groups**
      - ◆ **All > User Administration > Roles**

- **All > System Security > Users and Groups > Users**
- **All > System Security > Users and Groups > Groups**
- Bring up a **User** or **Group** record and assign roles under the *Roles* tab
  - Alternative: **All > System Security > Users and Groups > Roles**

- [ ] **Role** assignment and permissions hierarchy (roles within roles)
  - Assigning a role will include the permissions of that role and any roles contained within (ex. **catalog_admin** contains both **user_crtieria_admin** and **catalog**)
- [ ] Understand best practices for **Role** usage in the platform
  - Roles should **not** be assigned to users. Instead, create a Group, assign the role to the Group and add the required Users to the created Group to give them the role(s)

📝**Practice Tasks** [*SNAF Lab 2.4: Grant Access to Applications*]

- [ ] Add new **User** via **System Security** or **User Administration** modules
- [ ] Update **Group** membership
- [ ] Update **Role**s assigned to **Group**

➔ Common user interfaces in the platform

💡**Knowledge** [*Welcome to ServiceNow*]: (config covered in other modules)

- [ ] Lists
- [ ] Forms
- [ ] Tasks
- [ ] Dashboards
- [ ] Reporting
- [ ] Workspaces
- [ ] Knowledge
- [ ] Service Catalog
- [ ] Virtual Agent

## LD3: Config Apps for Collaboration (20%)

*MD3: Configure Applications for Business*

- ➔ Lists, Filters and Tags
  - ◆ **Filters / Context Menus / Tags**
    - 💡**Knowledge** [*SNAF Module 3.1 - Create Classic List Views*]
      - ☐ How to access a **List** and what it represents in both Classic View / Workspace
        - ◆ **List** - displays a list of records from a table
        - ◆ can be found on WS landing pages, WS UX Lists or in CV application modules
      - ☐ Classic List - Context Menus (3) and their usage
        - ◆ **List Control Menu** - display options related to viewing and filtering the list
        - ◆ **Column Options Menu** - display options specific to that column (reporting, configuring, export, etc.)
        - ◆ **Record Context Menu** - display options specific to that field (filtering, assigning, etc.)
      - ☐ Classic List - Filters / Quick Filters and their usage:
        - ◆ Click the **show/hide filter icon** to edit filter conditions
        - ◆ There are also quick filter options in the above context menus
      - ☐ Know the **3** components that make up a **Filter Condition**
        - ◆ **Field**, **Operator**, **Value**
      - ☐ Classic List - Inline editing / editing multiple records (Shift key)
        - ◆ Hold down the **Shift** key to select multiple records, or use checkboxes
        - ◆ To edit a single field on multiple list items, click column header and choose **Update Selected**
      - ☐ Classic List - Know the field right-quick Quick-edit functions (4) and how they appear
        - ◆ **Assign to me**
        - ◆ **Assign Tag**
        - ◆ **Approve** (for records that use approvals)
        - ◆ **Reject** (for records that use approvals)
      - ☐ Classic List - Configure List Layout
        - ◆ Click **Column Options Menu** and choose **Configure > List Layout**
        - ◆ This brings up a **List Collector** which allows you to:
          - add/remove/reorder fields on the **Selected** list
          - Change the list view, or create a new view
          - Create a new field for the table
      - ☐ List Collectors & Dot-Walking
        - ◆ **List collectors:** allow you to add/remove/reorder fields for a List
        - ◆ **Dot-Walking**:allows you to add fields from other related tables to the current view
      - ☐ Role(s) required to configure a List (add/remove/reorder) for all users
        - ◆ **personalize_list**
      - ☐ Adding new applications or modules via **Application Menus**
        - ◆ **All > System Definition > Application Menus**
      - ☐ Classic List - List Personalization
        - ◆ Click the **Personalize List** gear icon in the list column header to access
        - ◆ Use the **List Collector** to add/remove/reorder columns for the individual user
      - ☐ Tags
        - ◆ Right-click a field and choose **Assign Tag > New Tag** to tag a record
    - 📝**Practice Tasks** [*SNAF Lab 3.1 - Create and use List View - Classic*]
      - ☐ Navigate to **All > incident > All** and select **Configure>List Layout** on existing List
      - ☐ Create a new **View** and name it.
      - ☐ Use **List Collector** and **dot-walking** to add/remove/reorder fields
      - ☐ Save new **View** and verify it is visible from the List Context Menu
      - ☐ Add a new **Application Menu** item and **Save**
      - ☐ Add a new **Module** to display the new **List View** for the **Application** item created
      - ☐ Set *Visibility* and **Link Type* for new **Module** and verify new App/Module are visible in the **All** menu
- ➔ List and Form Anatomy
  - ◆ **List Views**

💡**Knowledge** [*SNAF Module 3.1 - Create Classic List Views*]:

- [ ] What is a **View** and how you switch between them
  - **view**: determines which fields to display and in what order
- [ ] How to create a new **View** for a **List**
  - Click **Column Options Menu** and choose **Configure > List Layout**, then under *List View* select **New…**
- [ ] What are the default **Sort** controls if none are defined
  - *order*, *number*, *name* or *[display field for the table]*
- [ ] Be aware that there are **List Views** that are linked to Workspaces but need to be edited in Classic View
  - Two examples are *Service Operations Workspace / SOW Landing Page*

- **Workspaces**

💡**Knowledge** [*SNAF Module 3.2 - Create Workspace List Views*]:

- [ ] Workspace List - Available options for viewing **Lists** (3)
  - Grid Mode, List Mode and Open List in New Tab
- [ ] Workspace List - Available context menus for **Lists** and what they do (3)
  - **Grab Icon**: allows you to drag/drop columns to reorder them (*Move Column*)
  - **More Options Menu**: group records that have the same values in a column (*Filter Column*)
  - **More UI Actions icon**: filter out records based on field values (*Show Matching/Filter Out*)
- [ ] Workspace List - Components that make up a **List Filter** (3) and how to access it
  - **Field**, **Operator**, **Value** - accessed by clicking the **show/hide filter** icon
- [ ] Workspace List - How to edit data, including multiple records
  - Hold down the **Shift** key to select multiple records, or use checkboxes
  - To edit a single field on multiple list items, click **Edit (#)** button make the change and click **Update**
- [ ] Workspace List - System property required to enable inline editing
  - **glide.lists.inline_editing_enabled**
- [ ] Workspace List - How to configure a **List Layout** in a Workspace using Classic View
  - Same as configuring a non-workspace list: **Configure > List Layout**
- [ ] Workspace List - Application used to add a new **Category** and **UX List**
  - **UI Builder**
- [ ] Workspace List - High-level steps (3) to configuring **Categories** and **UX List**s in **UI Builder**
  - 1. Configure List Category
  - 2. Configure a List and Add to Category
  - 3. Make List Visible
- [ ] Workspace List - How to configure a **UX List**
  - Open **UI Builder** and select **Service Operations Workspace**
  - Select:
    - **List** from *Page Options*,
    - **List nav** from *Content Panel* and
    - **Configuration** in the *Configuration Panel*
  - Open the correct **List Menu Configuration** record
  - There will be tabs for *UX List Categories* and *UX Lists* under *Related Links*
    - Create or edit the *Category* and create or edit the *List* (just ensure that the *List* specifies the correct *Category* to match
- [ ] Workspace List - Establishing **Audience** and **List Applicability** in **UI Builder**
  - Within **UI Builder** select **M2M Applicability** under *View* to bring up **List Applicabilities**.
  - Open the desired record (or create a new one)
    - set *Applicability* to the desired one (for ex. SOW ITIL Audience)
- [ ] Workspace List - List Personalization
  - Click the **List Actions** icon and select **Edit Columns**

📝**Practice Tasks** [*SNAF Lab 3.2 - Create and use List Views - Workspace*]

- [ ] Update **All > Incident > Open List View** for **SOW Landing Page** by adding a new field

- [ ] Save the changes to the **View** and verify with impersonation visible in Workspace
- [ ] Update **All > Incident > Open** List View for **Service Operations Workspace** view
- [ ] Save the changes to the **View** and verify with impersonation visible in Workspace
- [ ] Configure a new **UX List Category** using **UI Builder** for "***Default - SOW*"** record
- [ ] Configure a new **UX List** using **UI Builder** for "***Default - SOW***" record
- [ ] Verify that the new **Category** and **UX List** are both visible in the **Workspace list**
- [ ] Create a new **List Applicability** and link it to an **Audience** using the **UX List** created above
- [ ] Verify desired **Audience** is able to view the **Category** and **UX List** as desired
- [ ] Be aware that there are **List Views** that are linked to Workspaces but need to be edited in Classic View (ex. *Service Operations Workspace / SOW Landing Page*)

- **Forms**
  - 💡**Knowledge** [*SNAF Module 3.3 - Add Fields to Forms*]:
    - [ ] How to access a **Form** and what it represents in both Classic View and Workspace
      - a **Form** displays fields from a single record
    - [ ] The anatomy of a Form in both Classic View **(CV)** and Workspace **(WS)** View
      - **CV** form displays the following information:
        - Fields
        - Sections
        - Activity Stream
        - UI Actions (Related links)
        - Related lists
      - **WS** form displays the following information:
        - Form fields are located under *Details* tab
        - Form sections are grouped
        - *Activity stream* appears as an additional panel in the workspace

➔ Form Configuration

- **Form Design / Form Layout**
  - 💡**Knowledge** [*SNAF Module 3.3 - Add Fields to Forms*]:
    - [ ] Know to select the correct **View** before configuring a **Form**
      - **List Controls Menu > View >** [desired view]
    - [ ] Understanding of both **Form Design** and **Form Layout** configuration methods and how to select them
      - **Form Design** is a more visual drag-and-drop type interface
      - **Form Layout** is shown as a standard List Collector interface with **–split–** and **–end split–** used to determine which fields appear in which columns
      - Either can be found by right-clicking the header (or context control menu) and selecting *Form Layout* or *Form Design* respectively
    - [ ] Know what functionality is **not** available in a List Collector using **Form Design**
      - dot-walking
    - [ ] **Form Layout** - how to add a new **Field**
      - Add a new field using the *Create new field* dialog of the List Collector
    - [ ] **Form Layout** - how to edit **Related Lists** and their relationship to the current record
      - right-click the header and choose **Configure > Related Lists** or you can use the **New** and **Edit** buttons on the actual *Related Lists* section when viewing a record
    - [ ] How to **Personalize Form** in both Classic View and in Workspace view
      - In Classic View, click the **Personalize Form** icon and check/uncheck as desired
      - In Workspace, under the *Details* tab, click the **jump-to-section navigation icon** and select **Personalize Form**
    - [ ] Know the system property to disable **Personalize Form** for the itil role
      - set **glide.ui.personalize_form.role = admin**
    - [ ] **Form Design** - know what should be pre-selected prior to config
      - Application Scope
    - [ ] **Form Design** - main components of the form designer (3)
      - Page Header, Field Navigator, Form Layout
    - [ ] Understand what **Formatters** are and where they are used in the platform

- **formatter**: form element which displays information that is **not** a field in a record

📝**Practice Tasks** [*SNAF Lab 3.3 - Add Field to Form - Classic and Workspace*]

- [ ] **Configure > Form Layout** on an Incident, ensure **View** is correct
- [ ] Update the List Collector to add / rearrange fields
- [ ] Save and verify the update went through as expected.
- [ ] **Configure > Form Layout** to edit the Workspace View for **Service Operations Workspace**
- [ ] Make the same change as above, verify view is updated in Workspace as expected
- [ ] Make the same change as above but to the Workspace View **Service Operations Workspace New Record**
- [ ] Verify when creating a new record that desired changes are visible.

➔ Advanced Form Configuration

- **Choice / Reference Lists**

💡**Knowledge** [*SNAF Module 3.4 - Add Category and Reference Values*]:

- [ ] Understand what a **Choice List** is and how to use **Show Choice List** and **Configure Choices** options to customize the list
  - **choice list**: type of field that lets user select from a predefined set of choices
    - right-click the **field label** and choose **Configure Choices**
      - this brings up an interface that allows you to add/remove or create new choices for that specific table
    - right-click the **field label** and choose **Show Choice List**
      - this displays all Choices available for the chosen field across all tables of the same type (ex. Task)
- [ ] Understand what a **Reference List** is and how to add new items or manage existing
  - **reference list**: type of field that lets user select from a value from referenced records
    - Click the **magnifying glass icon** next to the field to bring up the list

📝**Practice Tasks** [*SNAF Lab 3.4 - Add Category and Reference Values*]

- [ ] Create a new **Group** to support a new **Service** and **Service Offering** for an Incident
- [ ] Use **Configure Choices** on a **Form** to add a new item to a Choice List for **Service**
- [ ] Create a new **Service** entry
- [ ] Create a child **Service Offering** under the *Offerings* tab
- [ ] Update the *Support group* and *Change group* fields to be the new group on the **Service Offering r**ecord.
- [ ] Update the *Support group* and *Change group* fields to be the new group on the **Service** record as well.
- [ ] Impersonate an **itil** user and verify that the new **Service** and **Service Offering** are available in the relevant **Choice Lists**. Ensure as well that *Assignment Group* auto-populates with the new Group created.

## LD3: Config Apps for Collaboration (20%)

*MD5: Enable Productivity*

➔ Reporting, Dashboard, Performance Analytics

- **Reporting**

💡**Knowledge** [*SNAF Module 5.1 - Create Reports and Dashboards*]

- [ ] How to access **Reports** and the benefits of using them (what do they show)
  - **All > Reports > View/Run**
  - Allows users to view and analyze ServiceNow data
- [ ] Know the different types of **Reports** available and what they do (**6** categories)
  - Bars, Pies and Donuts, Time Series, Multi-dimensional reports, Scores, Other
- [ ] Know the **Report** Visibility Controls available in the **Reports > View/Run** module
  - *My Reports*, *Group*, *Global*, *All*
- [ ] Know the **4** sections in the **Report Designer** and what they allow you to perform
  - 1. **Data**: name the report and select source type/source
  - 2. **Type**: choose the type of report (Pie, Bar, etc.)
  - 3. **Configure**: grouping options/data calculations
  - 4. **Style**: adjust look of the report (colors/legend)
- [ ] Know the **4 Save** options for a Report
  - 1. **Save**: Update the report and stay there
  - 2. **Update**: Update the report and return to the report list
  - 3. **Insert**: Save a duplicate copy of the report and return to the report list
  - 4. **Insert and Stay**: Save a duplicate copy of the report and remaining there
- [ ] Know the **5** options under the **Sharing** menu:
  - 1. **Share** - share the report, specify users/groups
  - 2. **Schedule** - set the report to generate at a specific date/time
  - 3. **Export to PDF** - export a copy of the report to .PDF format
  - 4. **Add to Dashboard** - add the report to a Dashboard
  - 5. **Publish** - create a public URL to share a copy of the report to
- [ ] Understand how to add a report to a **Dashboard**
  - can add from Dashboard view, or directly from the report via **Sharing** menu

📝**Practice Tasks** [*SNAF Lab 5.1 - Create Reports and Dashboards*]

- [ ] With an existing List of records displayed, generate a new **Report** in **Report Designer**
- [ ] Configure the **Report** using the **Data, Type, Configure** and **Style** sections and **Save**
- [ ] Locate the report in **Reports > View/Run** module and favorite it.
- [ ] Re-open the report and set **Sharing** for the report for a user and/or groups.
- [ ] Impersonate the relevant user and verify they are able to see the report.

- **Dashboards**

💡**Knowledge** [*SNAF Module 5.1 - Create Reports and Dashboards*]

- [ ] How to access a **Dashboard**
  - **All > Self-Service > Dashboards**
- [ ] How to share a **Dashboard** with specific users/groups
  - Click the **Sharing** menu and specify Groups, Users or Roles to **Can View** or **Can Edit**
- [ ] How to configure widgets within a **Dashboard**
  - Click **Add Widgets**, choose the designer widget and click **Add**
- [ ] How to launch **Interactive Analysis** and awareness of it's features
  - while viewing a List of records, click the **column options menu** and choose **Launch Interactive Analysis**
  - Click the **Filter Info** icon to edit the Source Filter or get a Share link
  - Click the **Filters** icon to further filter results
- [ ] How to share the link to **Interactive Analysis** with other users
  - Click the **Filter Info** icon and copy the URL linked under SHARE

📝**Practice Tasks** [*SNAF Lab 5.1 - Create Reports and Dashboards*]

- [ ] Create a new **Dashboard** at **All > Self-Service > Dashboards**.
- [ ] Add the above **Report** to the **Dashboard**
- [ ] Update **Dashboard** to share it with specific users/groups
- [ ] Impersonate the relevant user and verify they are able to see the dashboard.

- [ ] Launch **Interactive Analysis** from a **List** from the Column Context menu to explore

- **Tags**
  - 💡**Knowledge** [*SNAF Module 5.1 - Create Reports and Dashboards*]
    - [ ] How to use **Tags** within a **List** (2 methods)
      - right-click the **record context menu** and choose **Assign Tag**
      - You could also add the *Tag* column and use in-line editing
    - [ ] How to use **Tags** within a **Form** (in both Classic & Workspace views)
      - **Classic view**: click **More Options** and choose **Add Tag**
      - **Workspace view**: click the tag icon next to the primary value to add/edit
    - [ ] Know the **3** options for the **Viewable by** field and the role required for *Everyone*
      - *Me*, *Groups and Users*, *Everyone*
- **Performance Analytics**
  - 💡**Knowledge** [*SNAF Module 5.1 - Create Reports and Dashboards*]
    - [ ] How to access **Performance Analytics** dashboard
      - **All > Self-Service > Dashboards,** choose *ServiceNow Performance*
    - [ ] Know that this provides information about performance **over time** (iteratively)
    - [ ] Know the base **Performance Analytics** architecture
      - Available *Graph Sets*:
        - **AMB**, **Database**, **Instance View**, **MySQL Global Status**, **ServiceNow Servlet**, **Slow Pattern**
      - Can set item to **monitor** (instance), **Timespan** and do **graph comparisons**
      - Some graphs can be filtered out of categorized results by clicking on them
    - [ ] Understand the difference between standard **Reporting** and **Performance Analytics**
      - **Reporting** generates a report about the current state (*Where are we now?*)
      - **Performance Analytics** is more about reviewing historic trends and forecasting the future (*Where have we been?* and *Where are we going?*)

➔ Form templates and saving options

- **Form Templates**
  - 💡**Knowledge** [*SNAF Module 5.2 - Form Templates (*** bonus ***)*]
    - [ ] Know what a **Form Template** is and how to create one
      - **form template**: allows form fields to be automatically populated, simplifying creating a new record
      - Create one by clicking **More Options**, choosing **Toggle Template Bar** (if not already visible) and then the + icon
    - [ ] Understand how to apply a **Form Template** to a form (ex. Incident)
      - Apply one by clicking **More Options**, choosing **Toggle Template Bar** (if not already visible) and then click the name of the template you want to apply
    - [ ] Understand which role grants the ability to create and/or share **Form Templates**
      - The role of **template_editor_group** needs to be added
  - 📝**Practice Tasks** [*SNAF Lab 5.2 - Create Form Templates (*** bonus ***)*]
    - [ ] Open an Incident and create a new **Form Template**.
    - [ ] Update the **Groups** field to reflect a specific group who should see/use the template
    - [ ] Review fields on the template form and tailor them as desired
    - [ ] Save the template and apply to the current record
    - [ ] Impersonate the relevant user of a group and verify template is available to them.
    - [ ] Grant the **template_editor_group** role to the above user and verify they can add and share new templates.

➔ Task Management

- 💡**Knowledge** [*Welcome to ServiceNow*]
  - [ ] How to manage and assign **Task-based** records
    - Tasks are assigned to *Assignment Groups* and once assigned to an individual, will have an *Assigned to* filled in to reflect the User assigned
  - [ ] Know how to access the **My Work** and the **My Group's Work** modules
    - **All > Service Desk > My Work**
    - **All > Service Desk > My Group's Work**
  - [ ] Understand how **User Presence** aids in task completion and collaboration

- Allows you to see who else is looking at or working on the same task, in addition to whether they are available, logged in or away

- [ ] Know how to identify fields recently updated by other users
  - These will show with a **pulse icon** next to the field that has been updated
- [ ] Understand how to add Work Notes/Comments and review the Activity Stream
  - covered already in **LD1** section above, under **ITSM Incident Management**

## ➔ Visual Task Boards (VTBs)

### 💡**Knowledge** [*Welcome to ServiceNow*]

- [ ] How to access a **Visual Task Board** (from **All** menu or **List** view) and how they work
  - On a List, click the **Column Options Menu > Show Visual Task Board**
- [ ] Know the **4** different sections within a VTB and what they are for
  - 1. **Taskboard Tools**: filter tasks, VTB info, add/view members, add or filter labels, show activity stream and config
  - 2. **Quick Panel**: active participants or assignees
  - 3. **Lanes**:categories
  - 4. **Cards**: task records
- [ ] Know the **2** types of **VTB**s supported and the difference between each
  - 1. **free-form**: use as a personal organizer, completely customizable
  - 2. **data-driven**: tied to data, cannot change titles on lanes

## ➔ Quick Messages

### 💡**Knowledge** [*SNAF Module 5.3 - Enable Quick Messages (*** bonus ***)*]

- [ ] Understand how to access and use the email client from within a **Form**
  - Click the **More Options** menu and choose **Email**
- [ ] Understand how to access and use the email client from within a **Workspace**
  - Click the **More Options** menu and choose **Compose Email**
- [ ] Know how to access Email Client Templates from **All** menu
  - **All > Email Client > Email Client Templates**
- [ ] Know how to access and create a **Quick Message**
  - **All > Email Client > Quick Messages**
- [ ] Know the various fields you can populate during creation and how to insert variables/data into the body of the message
  - *Title*, *Active*, *Application*, *User*, *Group*, *Table*, *Conditions*, *Body*
  - standard syntax to reference field values: **{table_name.variable_name}**
- [ ] Know how to insert a **Quick Message** into an email
  - Within the **Email Client**, use the *Quick Message* selector
  - By default, content is inserted at the place of the cursor

### 📝**Practice Tasks** [*SNAF Lab 5.3 - Enable Quick Messages (*** bonus ***)*]

- [ ] Add the **email_client_quick_message_author** role to desired User
- [ ] Impersonate the User and create a new **Quick Message**
- [ ] Add text for the quick message and use embedded variable names to use dynamic content based on the Record
- [ ] Save the **Quick Message**.
- [ ] Create a new record and insert the **Quick Message** in Workspace view
- [ ] Complete email, send and verify content with admin in **System Mailboxes - Outbox**
- [ ] Create a new record and insert the **Quick Message** in Classic view
- [ ] Complete email, send and verify content with admin in **System Mailboxes - Outbox**

## ➔ Notifications

### 💡**Knowledge** [*SNAF Module 5.4 - Configure Notifications*]

- [ ] Understand what a **Notification** is and what **three** methods can be used to send
  - **notification**: a tool for alerting users when events that concern them have occurred through:
    - 1. Email
    - 2. SMS
    - 3. Meeting Invitation
- [ ] Know **All** menu path to access **Notifications**
  - **All > System Notifications > Email > Notifications**

- [ ] Know **All** menu path to access **Email Layout** and **Email Template**
  - **All > System Notifications > Email > Templates**
  - **All > System Policy > Email > Layouts**
- [ ] Know the **three** steps to create an **Email Layout** and apply to **a Notification**
  - 1. Create Email Layout
  - 2. Apply Layout to Template
  - 3. Apply Template to Notification (*What it will contain* tab)
- [ ] Know the **three** tabs required to create a **Notification**
  - 1. *When to send*
  - 2. *Who will receive*
  - 3. *What it will contain*
- [ ] Know **four** options available for **When to send**
  - 1. Record inserted or updated
  - 2. Event is fired
  - 3. Triggered
  - 4. Script (when in advanced view mode)
- [ ] Understand what **Watermark** labels are used for and how they are formatted
  - **Used for:** allows the platform to match incoming email to existing records. If watermarks are omitted, inbound actions may not work properly.
  - **Format**: **31** characters:
    - **3** - customizable prefix
    - **7** - source record identifier
    - **21** - underscore and random 20-char string

📝**Practice Tasks** [*SNAF Lab 5.4 - Configure Notifications*]

- [ ] Create a new **Notification**
- [ ] Update *When to Send* tab to trigger based on certain parameters (ex. Priority = 1)
- [ ] Update *Who will Receive* tab to a user without hard-coding their name (dynamic)
- [ ] Update *What it will contain* tab to detail the message, including variables
- [ ] Click **Preview** to verify everything looks correct, then save the **Notification**.
- [ ] View all **Email Notifications** and add *Mandatory* and *Force Delivery* columns.
- [ ] Set both *Mandatory* and *Force Delivery* columns to **true**
- [ ] Create a new record/trigger conditions to test the Notification was triggered as expected in the **Outbox**

➔ Sidebar

💡**Knowledge** [*SNAF Module.4 - Configure Notifications*]

- [ ] Understand what **Sidebar** offers for functionality and collaboration
  - Opens a real-time collaboration and discussion pertaining to a task-based or interaction-based record with other users.
- [ ] Know that **Sidebar** can have multiple docked windows, and shows User Presence
- [ ] Know **Sidebar** is only available in **Tokyo** or later releases
- [ ] Know that **Sidebar** messages can be added to the **Activity Stream** in a record

➔ Integration Hub

💡**Knowledge** [*SNAF Module 5.4 - Configure Notifications*]

- [ ] Basic overview of what **Integration Hub** is used for and the concept of **spokes**

➔ Automation Center

💡**Knowledge** [*SNAF Module 5.4 - Configure Notifications*]

- [ ] Overview of the **Automation Center** and the benefits it provides (ex. RPA integration)

➔ Predictive Intelligence

💡**Knowledge** [*SNAF Module 5.4 - Configure Notifications*]

- [ ] Overview of what **Predictive Intelligence** is and how it improves efficiency and quality

## LD3: Config Apps for Collaboration (20%)

*MD6: Manage an Instance*

- ➔ Performance Dashboard
  - 💡**Knowledge** [*SNAF Module 6.1 - Identify High Volume and Slowness Patterns*]
    - ☐ Understand the importance of Change Management and the basic stages involved
      - ◆ Changes should be planned and must fit into pre-defined requirements including:
        - Scheduling the change
        - Understanding blackout windows
        - Coordinating implementation with other changes to reduce conflicts
      - ◆ Change requests move through a basic process which includes:
        - Creation and Scope
        - Approval
        - Implementation
        - Closure
    - ☐ Understand how to access the **Performance Dashboard**
      - ◆ **All > Self-Service > Dashboards,** choose *ServiceNow Performance*
    - ☐ Know the base **Performance Analytics** architecture
      - ◆ Available *Graph Sets*:
        - **AMB**, **Database**, **Instance View**, **MySQL Global Status**, **ServiceNow Servlet**, **Slow Pattern**
      - ◆ Can set item to **monitor** (instance), **Timespan** and do **graph comparisons**
      - ◆ Some graphs can be filtered out of categorized results by clicking on them
  - 📝**Practice Tasks** [*SNAF Lab 6.1 - Identify High Volume and Slowness Patterns*]
    - ☐ Open the **Performance Dashboard**
    - ☐ Set the **Graph Set** to *Instance View* and **Timespan** to 30 days
    - ☐ Look at **Transaction per minute** and **Response times** graphs, hover over data
    - ☐ Set the **Graph Set** to *ServiceNow Servlet* and hover over data
    - ☐ Using the data and filters at the bottom, identify the **Network** active threads
- ➔ System Diagnostics
  - 💡**Knowledge** [*SNAF Module 6.1 - Identify High Volume and Slowness Patterns*]
    - ☐ The **System Diagnostics** page is deprecated since **San Diego** release
    - ☐ Know the **Stats Tools** availability and remember **stats.do**!
- ➔ Instance Scan
  - ◆ **Checks / Suites / Findings / Cleanup**
    - 💡**Knowledge** [*SNAF Module 6.2 - Monitor an Instance (*** bonus ***)*]
      - ☐ How to access **Instance Scan** from the **All Menu** (or **Dashboards**)
        - ◆ **All > Instance Scan > Dashboard**
        - ◆ Or find it under **All > Self-Service > Dashboards**
      - ☐ Understand what **Checks** are and the **4** types that are available
        - ◆ **checks**: singular focused roles that detect anomalies or opportunities in an instance
          - Table check
          - Column Type check
          - Script Only check
          - Linter Check
      - ☐ Understand what **Check** scores of 100 and <100 indicate
        - ◆ 100 indicates the check passed and there will be no record on the findings list
        - ◆ <100 indicates that there are exceptions and there will be a record on the findings list
      - ☐ Understand the concept of **Suites**
        - ◆ **Suites:** group of individual **checks** and suites associated together
      - ☐ Understand what **Results** and **Findings** are and how they map to each other
        - ◆ **result**: result from performing the scan
        - ◆ **finding**: reference to a record that has violated a rule from a check
      - ☐ Understand what the **Table Cleanup** option does
        - ◆ erases previous scan results
    - 📝**Practice Tasks** [*SNAF Lab 6.2 - Monitor and Troubleshoot Instance (*** bonus ***)*]
      - ☐ Navigate to **Instance Scan > Suites** and select one to see the available **Checks**

- [ ] Navigate to **Instance Scan > Checks** and click **Execute Full Scan**.
- [ ] Once completed, click **Go To Result** and review the scores on the **Checks** for any that are less than 100.
- [ ] Click on the *Scan Findings* tab and select the relevant <100 **Check** to view details
- [ ] Read the *Description* and *Resolution* sections and take any necessary actions
- [ ] Investigate the **Instance Scan** dashboard and see any historic changes since the previous scan.
- [ ] Use **docs.servicenow.com** to find differences between **Instance Scan** and **HealthScan** and research **Instance Troubleshooter** as well

➔ Health Scan

💡**Knowledge** [*SNAF Module 6.2 - Monitor an Instance*]

- [ ] Be aware that this tool exists and watch the Youtube video to learn more

➔ IT Adoption Journeys

💡**Knowledge** [*SNAF Module 6.3 - Extending Value*]

- [ ] Know how to access and be familiar with what **IT Adoption Journeys** provide
  - Under **Admin** in top header -> **IT Adoption Journeys**
  - Provides a central hub for admins to discover platform capabilities and discover new applications.
- [ ] Be aware of the **Business Objectives** and how to view **Adoption Blueprints**

➔ Upgrades and Releases

💡**Knowledge** [*SNAF Module 6.3 - Extending Value (*** bonus ***)*]

- [ ] How to access **Release Notes** and **Upgrades** help
  - **docs.servicenow.com** > *Release Notes and Upgrades* section
- [ ] Know how to assemble and build custom **Release Notes**
- [ ] Know what a **Family** is and the difference between **Upgrading** and **Updating**
  - **family**: set of releases named after a major city (ex. Tokyo)
  - **upgrading**: act of moving to a release in a different family
  - **updating**: act of moving from one patch/hotfix to another within the same family
- [ ] Know the **7 Upgrade Phases** and what they entail
  - 1. Read release notes and plan upgrade
  - 2. Prepare for the dev instance upgrade
  - 3. Verify upgrade configurations and schedule the dev instance upgrade in ServiceNow support
  - 4. Upgrade and validate the dev instance
  - 5. Upgrade and validate your other non-prod instances
  - 6. Prepare to upgrade the prod instance
  - 7. Upgrade the prod instance
- [ ] Know how to access the **Upgrade Center**
  - **All > Upgrade Center**
- [ ] Know what is entailed to perform a **Rollback** and the available window for support
  - ServiceNow does **not** provide a universal rollback option
  - Rollback window is 10 days by default, contact ServiceNow Support

📝**Practice Tasks** [*SNAF Lab 6.3 - Locate Release and Upgrade Resources (*** bonus ***)*]

- [ ] Go to **docs.servicenow.com** and find the **Release notes and upgrades** panel for your current version.
- [ ] Find and click the option to “Build your own release notes.”
- [ ] Click on “Release note summaries.”
- [ ] Select ”highlights for all Tokyo features and products.”
- [ ] Filter on **Application or Feature**.
- [ ] Click the **Download** icon and choose **Save as PDF.**
- [ ] Bring up the **Upgrade Planning Checklist** and review the **Prepare your Upgrade** section

## LD4: Self Service and Automation (20%)

➔ Knowledge Management

- **Architecture**
  - 💡**Knowledge** [*SNAF Module 4.1 - Knowledge Management*]:
    - ☐ Understand what **Knowledge Management** is and how to access it
      - **Knowledge Management**: Allows users to create, categorize, review, approve and browse important information in a centralized location , divided into one or more knowledge bases
      - **All > Self-Service > Knowledge**
    - ☐ Know the **role** required to administer Knowledge Bases
      - **knowledge_admin**
    - ☐ Be familiar with the **Guided Setup** for first-time setup of **Knowledge Mgmt**
    - ☐ Know how to create a new **Knowledge Base** without **Guided Setup**
      - **All > Knowledge > Administration> Knowledge Base**
    - ☐ Understand the architecture and organization of articles
      - Articles are organized into categories
      - **Only one** article can be associated to a Knowledge Base
      - Many categories can be associated to a Knowledge Base
- **User Criteria / Workflows**
  - 💡**Knowledge** [*SNAF Module 4.1 - Knowledge Management*]:
    - ☐ **W**hat **User Criteria** is and how it is defined in the platform
      - defines conditions that are evaluated against users to determine which users can create, read, write or retired knowledge articles
    - ☐ Know the **4** basic outcomes when managing access through **User Criteria** records
      - 1. canRead
      - 2. cantRead
      - 3. canContribute
      - 4. cantContribute
    - ☐ Know the System Property used to ensure all logged in users have access to KBs.
      - **glide.knowman.block_access_with_no_user_criteria**
    - ☐ Know the **6 Workflows** available in baseline for publishing and retirement process as provided in training:
      - 1. Approval Publish
      - 2. Approval Retire
      - 3. Instant Publish
      - 4. Instant Retire
      - 5. Publish Knowledge
      - 6. Retire Knowledge
    - ☐ Know that there is functionality to import from **Word** into an article
      - **All > Knowledge > Articles > Import Articles**
    - ☐ Be familiar with how to use Now Mobile to find **Knowledge** information
      - Use **All > Now Mobile App > Knowledge Bases** to see KBs visible in app
  - 📝**Practice Tasks** [*SNAF Lab 4.1 - Manage Knowledge Base and Create Articles (*** bonus ***)*]
    - ☐ Navigate to **All > Knowledge > Administration > Knowledge Bases**
    - ☐ Create a new **Group** which will contain the individuals to be assigned the **knowledge_admin** role
    - ☐ Add users to the group and assign both **knowledge_admin** and **approver_user** roles to the group
    - ☐ Select a KB and change the *Owner* to relevant User.
    - ☐ Change the *Publish Workflow* to **Knowledge - Approval Publish**
    - ☐ Change the *Retire Workflow* to **Knowledge - Approval Retire** and **Save**.
    - ☐ Update the *CanContribute* tab to reflect the new group created above.
    - ☐ Impersonate a user without the **knowledge_admin** role and use them to create a new **Article** in the Knowledge Base (import a Word doc optionally to do it)
    - ☐ Click **Publish** which will put the article into pending approval with KB Managers group

- [ ] Impersonate KB Manager, navigate to **All > Self-Service > My Approvals** and review article for accuracy and **Approve**
- [ ] Confirm the desired article is visible in the knowledge base.
- [ ] Create a new **User Criteria** record and add it to the *Can Read* tab of the Knowledge Base and **Save**.
- [ ] Impersonate a user **not** relative to the **User Criteria** created and verify they are not able to read the article
- [ ] Impersonate a user related to the **User Criteria** created and verify they **are** able to see the article.

➔ Service Catalog

💡**Knowledge** [*SNAF Module 4.2 - Create a Catalog Item*]:

- [ ] Know what the **Service Catalog** is used for and how to access it
  - **Service Catalog:** Robust ordering system for services and products
  - Accessed at **All > Self-Service > Service Catalog**
- [ ] Know the roles required to manage a **Service Catalog**
  - **catalog_admin**
- [ ] Know how to use the **Catalog Builder**
  - **Catalog Builder:** Create, edit and maintain catalog items (or record producers) using a visual and guided experience
  - **All > Self-Service > Service Catalog > Catalog Builder**
- [ ] Understand **Request Management** and the resulting elements created
  - Once an order has been placed, these types of items will be created:
    - **REQ** - Request, **RITM** - Request Item, **SCTASK** - Catalog Task
- [ ] Know how to access and edit Catalog Items and Variables
  - **All > Service Catalog > Catalog Definitions > Maintain Items**
- [ ] Understand all **5** major components of the **Service Catalog**
  - 1. Catalog Items
  - 2. Variables
  - 3. Variable Sets
  - 4. Record Producers
  - 5. Flows
- [ ] Know what a **Record Producer** is used for and specifically the type of record it is meant to produce
  - **Record producer**: simplified form that allows user to enter information that is translated into **Task**-based records
- [ ] Know the benefit of using an **Order Guide**
  - **Order guide:** provides the ability to order multiple, related items as a single request
- [ ] Know table prefixes and system tables for Request, Request Item and Catalog Task
  - **Request** - REQ - sc_request
  - **Request Item** - RITM - sc_req_item
  - **Catalog Task** - SCTASK - sc_task
- [ ] Know the **3** standard Flow stages for an item indicating the item's progress or state
  - *Waiting for Approval*, *Request Approved*, *Completed*
  - Others within these 3 could be: *Approved*, *Pending*, *Fulfillment*, *Deployment / Delivery*
- [ ] Understand how to create and use **User Criteria** to restrict access to catalog items
  - Apply a **User Criteria** record to either the item or category from the **item**, **category** or **user criteria** form

📝**Practice Tasks** [*SNAF Lab 4.2 - Create a Catalog Item*]

- [ ] Select a user and add him to the **Service Catalog** as an *Editor*
- [ ] Go to the user's account and add the **catalog_admin** role.
- [ ] Impersonate user and initiate the **Catalog Builder** using the standard template
- [ ] Populate forms as desired and be sure to add at least 3 questions, with 2 having multiple **Choices** to select from. (ex. Color / Size)
- [ ] Once at the **Review and Submit** page, Click **Preview** to confirm accuracy and **Submit** to finalize and add the catalog item.
- [ ] Navigate to **All > Service Catalog > Catalog Definitions > Maintain Items**

- [ ] Edit one of the Choices created above to have a higher price, ensure the item is set to **Active** and click **Try It** to verify.

➔ Flow Designer

💡**Knowledge** [*SNAF Module 4.3 - Flow Designer*]:

- [ ] Know what the **Flow Designer** is used for and how to access it
  - **Flow Designer**: Non-technical interface for building and enabling process automation capabilities, known as **flows**
  - **All > Process Automation > Flow Designer**
- [ ] Know the **3** roles required to work with **Flow Designer**
  - 1. flow_designer
  - 2. flow_operator
  - 3. action_designer
- [ ] Be familiar with the **Flow Diagram View** and the **5 trigger types** supported
  - 1. Record triggers
  - 2. Date triggers
  - 3. Inbound email
  - 4. Service Catalog
  - 5. SLA Task
- [ ] Know the **4** main components of a Flow:
  - 1. Triggers
  - 2. Conditions
  - 3. Actions
  - 4. Data
- [ ] Know the **3 types** of **Triggers**:
  - 1. Record-based triggers
  - 2. Date-based triggers
  - 3. Application-based triggers
- [ ] Understand how you specify **Conditions** for each **Trigger**
  - **Conditions**: statements that determines when or how an **action** runs
- [ ] Know the types of **Actions** available (especially **Service Core** actions)
  - *Ask for Approval*, *Create Record*, *Delete Record*, *Look up Record*, *Wait for Condition*
- [ ] Understand how to pull **Data** in using the **Data Pill Picker** icon (and dot-walking)
- [ ] Know the basic concept and utility for the **Process Automation Designer**
  - enables owners to author cross-enterprise workflows and create a single unified process

📝**Practice Tasks** [*SNAF Lab 4.3 - Create a Flow Designer Flow*]

- [ ] Select a **Catalog Item** and personalize to add the Flow column
- [ ] Review the selected Flow to understand the high-level actions.
- [ ] Create a new Flow
- [ ] Add and define a **Trigger** for *Service Catalog*
- [ ] Add multiple actions using **Data** where relevant (lab has you do the following:
  - **Update RITM Record** to set *State* to **Pending**
  - **Create Catalog Task** with desired *Assignment Group* and *State* to **Open**
  - **Send Email** to *Requested Item Record* > *Requested For* > *Email* and **CC** the *Requested Item Record* > *Requested For* > *Manager* > *Email*
    - populate subject/body as desired (ex. Add Requester First Name)
  - **Update RITM Record** to set *State* to **Closed Complete**
  - **Update REQ Record** to set *State* to **Closed Complete**
- [ ] **Save** the Flow (optionally **Test**) and **Activate** it
- [ ] Associate the created Flow to the selected **Catalog Item** by editing the *Process Engine* tab accordingly
- [ ] Impersonate a self-service user to test order the selected **Catalog Item**
- [ ] Step through the stages, impersonating required users as needed to complete stages where needed (approvals,etc.). Verify Flow works as desired, and check **Outbox** to verify if **Email** is involved.

➔ Virtual Agent

💡 **Knowledge** [*SNAF Module 4.3 - Flow Designer* ]:

- [ ] Know what the **Virtual Agent** does and what information it can access
  - **Virtual Agent**: conversational platform that helps users obtain information, make decisions and perform common working tasks within a messaging interface
- [ ] Know the **7** types of support tasks **Virtual Agent** can automate
  - 1. Answering FAQs
  - 2. Providing tutorial “how to” information
  - 3. Querying or updating records
  - 4. Gathering data, such as attachments, for the agent
  - 5. Performing diagnostics
  - 6. Resolving multi-step problems
  - 7. Working with file attachments during live chat

## LD5: Database Management (27%)

➔ Data Schema

- **Tables, Records and Fields**
  - 💡**Knowledge** [*SNAF Module 7.1 - Data Schema]*:
    - ☐ Know what **3** elements the Servicenow Infrastructure includes
      - 1. Tables
      - 2. Records
      - 3. Fields
    - ☐ Know how to access the **System Dictionary** and how records are identified
      - **All > System Definition > Dictionary**
      - Records are identified by a **sys_id**
    - ☐ Know how to access **Tables & Columns**
      - **All > System Definition > Tables & Columns**
    - ☐ Know the **3** key attributes for a **field**
      - 1. Name
      - 2. Label
      - 3. Value
    - ☐ Know what **Reference Fields** are
      - **reference field:** stores a sys_id of a record on another table which is what established the reference relationship
        - (ex. *Caller* is a **reference** to a record on *User* table)
- **Relationships**
  - 💡**Knowledge** [*SNAF Module 7.1 - Data Schema]*:
    - ☐ Know all **4** forms of **Table Relationships**
      - 1. One-to-many
      - 2. Many-to-many
      - 3. Database View
      - 4. Extensions
    - ☐ Know the **3** One-to-many relationship fields
      - Reference
      - Glide List
      - Document ID
    - ☐ Know Table **Parent** and **Child** Relationships, and how **inheritance** works
      - **Child**: extends from the parent table and **inherits all fields**
      - **Parent**: the table from which the child is extended from
    - ☐ Understand what **Base** tables are and **Core** vs. **Custom** tables
      - **Base:** table that can have child relationships, but no parent relationships
      - **Core**: table that exists in the the ServiceNow base system and can have both parent and child relationships
      - **Custom**: tables that are created by admins or developers
    - ☐ Know the namespace identifiers that prefix new tables
      - Unscoped (global) application: **_u**
      - Scoped application: **_x**
- **Schema Map**
  - 💡**Knowledge** [*SNAF Module 7.1 - Data Schema]*:
    - ☐ Know how to access the **Schema Map** and how to read the table relationships shown
      - 1. **All > System Definition > Tables**
      - 2. Open desired table, scroll down to *Related Links*
      - 3. Select **Show Schema Map**
  - 📝**Practice Tasks** [*SNAF Lab 7.1 - Create Table for HHD Configuration Records*]
    - ☐ Create a new **Table** in **Global Scope**:
      - *Label*: **Holographic Handheld HHD**
      - *Name:* u_**cmdb_ci_hardware_hhd**
      - *Extends table*: **cmdb_ci_hardware**.
      - *Add module to menu*: **HHD**
      - **Submit** to complete creation.
    - ☐ Configure the **HHD Module**'s list of records with desired columns (make the view unique by adding/removing columns, create a new field, etc.)

- [ ] Click **New** to display a new record form and select **Configure > Form Layout** to build a new form that will use the new Table
- [ ] Update the **Application Menu** to create a new Module::
  - *Link Type*: **New Record**
  - *Table*: **u_cmdb_ci_hardware_hhd**
  - *Title*: **Create New**
  - *Application Menu*: **HHD**
  - *Order*: **100**
  - click **Submit** to complete
- [ ] Submit a new record to test the new **HHD** Module, verify data flows to new table

➔ Application / Access Control

- **Access Controls**

💡**Knowledge** [*SNAF Module 7.2 - Application/Access Control]*:

- [ ] Know the **3** levels of security before an end user can perform CRUD table operations
  - 1. User Authentication (Users/Groups/Roles)
  - 2. Application/Modules access (Roles)
  - 3. Database access (Tables, Global System Properties, Access Control Rules)
- [ ] Know the **3** security modules typically used by the System Administrator
  - 1. **All > System Properties > Security**
  - 2. **All > System Security > Access Control (ACL)**
  - 3. **All > System Security > High Security Settings**
- [ ] Understand what an **Access Control** is and how it can be set
  - **access control**: security rule defined to restrict the permissions of a user from viewing and interacting with data
  - can be set at **row-level** (record) or **column-level** (field)
- [ ] Know how to navigate to and manage the **Access Control List**, which role is required
  - **All > System Security > Access Control (ACL)**
  - requires **security_admin** role
- [ ] Know what **3** elements defines an **Access Control**
  - 1. Operation
  - 2. Object
  - 3. Permissions
- [ ] Know how to view **Access Controls** for a table
  - Type **<table name>.config** into Filter Navigator, view *Access Controls* tab
- [ ] Know what **4 Access Controls** get generated by default when custom table is created
  - 1. **Create** access control
  - 2. **Read** access control
  - 3. **Write** access control
  - 4. **Delete** access control
  - Also, a role is created by default and associated with the access control rules
- [ ] Know the different **Rule Types** for Access Controls (**.none, .field, and .***) and how they work together to limit access (house analogy provided is helpful)
  - **table. –none–**: applies to whole table (whole house)
  - **table.field**: applies to a specific field (room in a house)
  - table.*: applies to a all fields without a table.field rule (all other rooms not defined by a house.field rules)
- [ ] Understand best practices for writing **ACL**s in terms of granting/denying access
  - When creating **.*** access controls, also create a **–none–** access control, because **only –none– grants access to records**
  - To mostly grant access, use **–none–**
  - To mostly deny access, use **–none–** and **.***
- [ ] Know the order **ACL rules** are processed
  - 1. Match object against table ACL rules - most specific to most general
  - 2. Match object against field ACL rules - most specific to most general

📝**Practice Tasks** [*SNAF Lab 7.2 - Create Access Controls*]

(*Section 1: Provide Application Menu and Module Access*)

- [ ] Navigate to **Roles** and verify new role for **u_holographic_handheld_hhd_user**
- [ ] Edit the **HHD Application**; remove **itil** and add the new role above

- [ ] Ensure that all Modules also have just this one role as well to limit visibility/access
- [ ] Navigate to the **Role** and verify you see both the Application and Modules associated
(*Section 2: Grant Roles to Support Groups*)
- [ ] Add the new role to **Service Desk** group record and to the **Training Technology Support Group**)
(*Section 3: Test the Visibility Settings*)
- [ ] Impersonate **Bernard Laboy** and verify they are not able to see the application or modules as desired
- [ ] Impersonate **Annette Frietas** and verify they ARE able to view and access both application and modules, and can complete a new form submission for **HHD**
(*Section 5: Establish Access Control Rules for Incident*)
- [ ] **Elevate** to **security_admin** to gain permission to manage **Access Controls**
- [ ] Navigate **to All > System Security > Access Control (ACL)** and create **3** access controls:
    - *Operation*: **read**
        - *Admin overrides*: **select the check box**
        - *Active*: **select the check box**
        - *Name*: **Incident [incident]**
        - *Description*: **u_holographic_handheld_hhd_user role required to read HHD incident records**
        - *Add filter condition*: **Service Offering | is | Infinity (HHD)**
    - *Operation*: **write**
        - *Description*: **u_holographic_handheld_hhd_user role required to write HHD incident records**
        - *Role*: **u_holographic_handheld_hhd_user**
    - Operation: **create**
        - Description: **u_holographic_handheld_hhd_user role required to create HHD incident records**
        - *Role*: **u_holographic_handheld_hhd_user**
- [ ] Search for all ***incident** at **All > System Security > Access Control (ACL)**
- [ ] Update the incident ACL with description: “itil role required to read incident records”:
    - *Description*: **itil role required to read incident records, except for Service Offering Infinity (HHD)**
    - *Add filter condition*: **Service Offering | is NOT | Infinity (HHD)**
- [ ] Update the incident ACL with description: “Allow read for records in incident…”:
    - *Description:* **Allow read for records in incident, for users with sn_incident_read, except for incidents for Service Offering Infinity (HHD)**
    - *Add filter condition*: **Service Offering | is NOT | Infinity (HHD)**
(*Section 7: Verify Access Control Rule for Incident read and write*)
- [ ] Impersonate **Bernard Laboy** and verify he sees a security constraint message at **All > Incident > Open** equal to the number of Infinity (HHD) incidents removed
- [ ] Impersonate **Beth Anglin** and verify she sees and can update all HHD incidents.
(*Section 8: Create an Access Control Rule for Asset Tag write*)
- [ ] Create a new **Role “u_hhd_asset_tagger” (**ensure **Global** scope):
- [ ] Navigate back to **All > System Security > Access Control (ACL)** and filter the list to find the four **u_cmdb_ci_hardware_hhd** rules
- [ ] Locate and open the **write** rule, select **Asset Tag** in the *Name* fields drop-down and select **Insert and Stay**
- [ ] Click past the **Verify Security Rules** window and add the **u_hhd_asset_tagger** role
- [ ] Click **Save** and **Update**. Verify the new **ACL** for **u_cmdb_ci_hardware_hhd.asset_tag** has been added in the list
(*Section 9: Grant the role for Updating Asset Tags*)
- [ ] Create a new **Group**:
    - *Group Name*: **HHD Asset Taggers**
    - *Role*: **u_hhd_asset_tagger**

- *User*: **Annette Frietas**

(*Section 10: Test the Asset Tag Settings*)

- [ ] Impersonate **Beth Anglin** and verify *Asset Tag* is read only on a new record
- [ ] Impersonate **Annette Frietas** and verify *Asset Tag* is writable on a new record

## ➔ Importing Data

### ◆ Import Datasets / Import Table

#### 💡**Knowledge** [*SNAF Module 7.3 - Import Data]*:

- [ ] Know what an **Import Set** is and the roles needed to manage them
  - **Import Set**: tool used to import data from various data sources, and map that data into ServiceNow tables
- [ ] Know what possible **5 Data Sources** are supported
  - CSV, XML, Excel, JBDC, HTTP/FTP
- [ ] Know what the **Import Set Table** and **Transform Maps** do
  - **Import Set Table**: acts as a staging area for imported records
  - **Transform Map**: provides a guide for moving data from Import Set tables to "Target" tables
- [ ] Be familiar with the **Automatic Mapping Utility** and the **Mapping Assist Utility**
  - **Automatic Mapping Utility**: simplest mapping method where all the field names of the Import Set match the name of the fields on the Target table
  - **Mapping Assist Utility**: provides a visually intuitive environment for specifying mapping between Import Set fields and Target table fields
- [ ] Understand why you would use **Coalesce** and how it functions as a unique key
  - **Coalesce**: allows you to specify a field as a **unique key**, meaning that an existing record will be updated as opposed to inserting and potentially creating duplicate records
- [ ] Know the **3** available **Coalesce** configurations
  - 1. Single-field
  - 2. Multiple field
  - 3. Conditional (based on a script returning sys_id of the target table record)

#### 📝**Practice Tasks** [*SNAF Lab 7.3 - Import Data*]

(*Section 1: Initial Data Load*)

- [ ] Go to **All > HHD > HHDs and Assignees** and make a new View "**Import Validation**"
- [ ] Open the new View and ensure that fields are ordered and match import excel file
- [ ] Navigate to **All > System Import Sets > Load Data,** complete form:
  - *Import set table*: **Create table** (auto-selected)
  - *Label*: **HHD Imports**
  - *Name*: **u_hhd_imports** (automatically populates)
  - *Source of the import*: **File** (auto-selected)
  - *File*: [choose Excel file **infinity_HHD-data.xlsx**]
- [ ] Click **Submit** and verify correct number of inserted records (should be **7**)
- [ ] Click the **Loaded Data** link under *Next Steps* and confirm all have *State* **Pending**
- [ ] Navigate to **All > System Import Sets > Create Transform Map**, complete form:
  - *Name*: **HHD CIs**
  - *Source table*: **HHD Imports** [u_hhd_imports]
  - *Target table*: **Holographic Handheld HHD** [u_cmdb_ci_hardware_hhd]
- [ ] **Save**, then click **Auto Map Matching Fields** under *Related Links* section
- [ ] Verify **2** fields are auto mapped: **Name** and **Asset Tag**
- [ ] Select **Mapping Assist** under *Related Links* section and drag-drop into *Fleld Map*:
  - Asset Tag | Asset tag
  - Name | Name
  - Device Number | Serial Number
  - Device Owner | Assigned to
  - Device Version | Firmware version
  - Owner Location | Location
- [ ] Click **Save** and verify there are now 6 mapped fields.
- [ ] Select **Transform** under *Related Links* and verify the correct **map** is selected.

- [ ] Click **Transform** and check the **Import Validation** view displays **7** records
  (*Section 2: Incremental Data Load*)
- [ ] Navigate to **All > System Import Sets > Load Data.** Complete the form:
  - *Import Set Table*: **Existing table** radio button
  - *Source of the import*: **File** (auto-selected)
  - *File*: [choose Excel file **infinity_HHD-updates.xlsx**]
- [ ] Click **Submit** and verify correct number of inserted records (should be **19**)
- [ ] Navigate to **All > System Import Sets > Administration > Transform Maps**
- [ ] Open the **HHD CIs** transform map and set *Coalesce* to **True** for **u_device_number**
- [ ] Under *Related Links* select the **Transform** link.
- [ ] Verify correct settings for *import set* and *selected maps*, then click **Transform**.
- [ ] Bring up the **HHDs and Assignees** list and verify **19** total records exist
  - Additionally, ensure that the *Updated* field correctly shows based on whether they existed in the initial import or are newer based on existing in the update import

  (*Section 3: Clean Up Import Set Tables*)
- [ ] Navigate to **All > System Import Sets > Import Set Tables > Cleanup** and complete:
  - *Delete these tables*: **HHD Imports [u_hhd_imports]**
  - *Delete related transform maps*: **unchecked**
  - *Delete data only (preserve table structure)*: **selected**
- [ ] Click **Cleanup**. Ensure you get a completed verification message and review the log.

➔ CMDB and CSDM

- **CMDB**

💡**Knowledge** [*SNAF Module 7.4 - CMDB]*:

- [ ] Know what the **CMDB** is, **CIs** and the **3 key system tables** pertaining to CMDB
  - **CMDB**: **C**onfiguration **M**anagement **D**ata**B**ase
    - 1. cmdb
    - 2. cmdb_ci
    - 3. cmdb_rel_ci
- [ ] Know how to access the **CMDB Workspace** and the information you can view there
  - Click **Workspaces** on top header and select **CMDB Workspace**
  - Allows you search and explore the CMDB, examine health and recent activity and access various CMDB dashboards and tools to support tasks in the organization
- [ ] Know what the **Configuration Item** form look like
  - divided into **CI Attributes** and **Relationships**
  - Can switch between **Dashboard** and **Form** view
- [ ] Familiarize with using the **CI Class Manager** and how to access it
  - **All > Configuration > CI Class Manager**
- [ ] Know the **3 CI Class Manager** Attributes tabs and what they filter
  - 1. **All** - all attributes
  - 2. **Derived** - all attributes derived from parent classes
  - 3. **Added** - all attributes that have been created specifically for the CI Class
- [ ] Know how to access the **CI Relationship Editor**
  - accessed from the *Related Items* toolbar

📝**Practice Tasks** [*SNAF Lab 7.4 - Explore Data Structures, CMDB and CI Relationships*]
(*Section 1: Explore Data Structures*)

- [ ] Go to **All > System Definition > Tables** and search for the **HR Case** table.
- [ ] Click **Show Schema Map** under *Related Links*. Verify **HR Case** extends from **Task**
- [ ] Show Schema Map for **Holographic Handheld HHD** table, see extension from multiple CMDB tables
  (*Section 2: Explore CI Class Manager - CMDB, CIs, and Relationships*)
- [ ] Navigate to **All > Configuration > CI Class Manager**, click **Open Hierarchy** and select the **Service** class. **Pin** it.
- [ ] Search for **Holographic Handheld HHD** class. **Pin** it.

- [ ] From **Pinned Classes** select **Service** class. Under **Class Info** click the **CI List** menu. Search for and select **Training**. Under *Related Items* click **Dependency Views** icon to view the **dependency map**.
  (*Section 3: Create New Relationship*)
- [ ] Navigate to **CI Class Manager** and select your pinned **Holographic Handheld HHD**. Navigate to **Suggested Relationships.**
- [ ] Add a new relationship as follows:
  - *Relationship*: **Runs** (Runs on::Runs)
  - *Target Class*: **Business Service**

  (*Section 4: Associate HHD devices with Service Offering*)
- [ ] Navigate to **All > Configuration > Service Offerings**. Select **Infinity (HHD)**.
- [ ] Click **+** on *Related Items* to access the Relationship Editor.
- [ ] From *Available Relationships* choose **Runs on…** and remove the filter condition for **Class**. Click **Run Filter**.
- [ ] Search **Infinity** devices in the *Available CIs* list then select and move them over to the "Runs on" section.
- [ ] **Save** and confirm the additions show up in the Service Offering relationships section.
  (*Section 5: View Relationship from HHD CI Record*)
- [ ] Go to the **Holographic Handheld HHD** CI in **CI Class Manager**.
- [ ] Edit the form to show *Legacy CI Relations* below *Comments*. Verify you are seeing the relationships.
  (*Section 6: Verify the Ability to use the CIs in the Incident Form (Classic*)
- [ ] Create a new Incident:
  - *Caller*: **Jon Floyd**
  - *Service*: **Training**
  - *Service Offering*: **Infinity HHD**
  - *Configuration Item*: Type **Infinity** and press the **magnifier** to look at the list. Pick the one that belongs to Jon.
  - *Pilot*: **checked**
  - *Short Description*:**Device fails to cast images**.
- [ ] **Save** and click on the **Tree** icon next to **Service** to view the hierarchy.

- **CSDM**

  💡**Knowledge** [*SNAF Module 7.4 - CMDB]*:

  - [ ] Understand what the **CSDM** model is
    - A set of terms and conditions governing a **CMDB-based framework**
  - [ ] Understand what the model defines, what it doesn't and what the benefits are
    - **What it is**: standard terms and definitions, best practice for CMDB Data, OOTB CMDB core tables, guidance on service modeling, recommended mappings
    - **What it is not**: a process or implementation guide for ITSM, SPM, APM, EM or Other products, not a set of reports, not code to install, not a SKU or product to buy, Not an automatic fix for past implementations
    - **Benefits**: consistently modeled service offerings, consistent use of service across applications, less overhead when maintaining services, product dependencies on CSDM tables

- **Discovery / Service Mapping**

  💡**Knowledge** [*SNAF Module 7.4 - CMDB]*:

  - [ ] Understand how **Discovery** works and how it finds devices and attributes on your network
    - **Discovery**: scans the network to inventory devices and applications and updates the CMDB with the results for each **unique type** of software and hardware
  - [ ] Understand how **Service Mapping** works and how it maps service-specific dependencies
    - **Service Mapping**: augments the CMDB with IT relationships and dependencies between CIs to model the IT components that comprise a Service
  - [ ] Understand the difference between **top-down mapping** and **horizontal discovery**
    - **Top-down mapping**: overlays service maps onto existing configuration data to connect CIs underlying a given service

- **Horizontal mapping**: creates an inventory of all devices and applications, but does not show how they are connected or dependent on each other

- **Dependency View**
  - 💡**Knowledge** [*SNAF Module 7.4 - CMDB]*:
    - [ ] Understand how the **Dependency View** depicts upstream and downstream relationships and the benefits of enhancing delivery of operational Incident, Change and Problem management process

## LD6: Data Migration and Integration (15%)

➔ Client Scripts

💡**Knowledge** [*SNAF Module 8.1 - UI Policies and Business Rules]*:

- ☐ Know what a **client script** is used for and the **4** types of scripts supported
  - **client script**: makes real-time changes to the appearance of the user interface, especially forms
  - 1. onCellEdit()
  - 2. onChange()
  - 3. onLoad()
  - 4. onSubmit()

➔ UI Policies

- **UI Policy and UI Policy Actions**

💡**Knowledge** [*SNAF Module 8.1 - UI Policies and Business Rules]*:

- ☐ Know what language scripting in **ServiceNow** is performed in
  - JavaScript
- ☐ Understand difference between executing script **client** or **server** side
  - **client-side**: web browser
  - **server-side**: ServiceNow database
- ☐ Understand what a **UI Policy** is
  - **UI Policy**: a rule that applied to a form to dynamically change information or the form itself
- ☐ Know the **3 UI Policy Actions** you can use with a **UI Policy**
  - 1. Mandatory or Optional
  - 2. Hidden or Visible
  - 3. Read-only or Editable
- ☐ Know what a **Data Policy** is and how it contrasts to a **UI Policy**
  - **Data Policy**: rule that enforces data consistency by setting fields as mandatory and/or read-only
  - **UI Policies** are only enforced on data entered into a form whereas **Data Policies** are enforced on **all** data entered into the platform
- ☐ Know that **UI Actions** can be executed either **server** or **client side**
  - This is dependent on whether or not the *client* checkbox is selected
- ☐ Be familiar with the **7** typical UI Actions available
  - Form buttons
  - Form context menus
  - Form links
  - List buttons
  - List context menus
  - List choices
  - List links

📝**Practice Tasks** [*SNAF Lab 8.1 - Create a UI Policy and Business Rules*]

(*Section 1: Prepare Forms*)

- ☐ Add **Contact** to top-right of new **Company** form (**User Administration > Companies**)
- ☐ Add **Manufacturer** to **Service Offering** under *Vendor*
- ☐ Prepare **Incident form**:
  - Create a new field for **Escalate to manufacturer**
  - Add the **Escalate to manufacturer** field to the **Incident** table (true/false)
  - Add the **Escalate to manufacturer** field to the **Incident** form, under the *Assigned to* field
- ☐ Add **Manufacturer contact** field under *Escalate to manufacturer* field:
  - *Name*: **manufacturer contact**
  - *Type*: **reference**
  - *Table*: **user**

(*Section 2: Prepare Data*)

- ☐ Create a **Test Contact** user record:
  - *User ID*: **Test Contact**
  - *First name*: **Test**
  - *Last name*: **Contact**

- *Email*: test.contact@test.com

- [ ] Create a **Company** record for **Infinity Products, Inc.** with *Contact* as **Test Contact**
- [ ] On the **Infinity (HHD) Service Offering**, add **Infinity Products, Inc.** as the *Manufacturer*

(*Section 3: Create UI Policy*)

- [ ] Create a New Incident and select **Configure > UI Policies.** Create **New** as follows:
  - *Table*: **Incident [incident]** (already selected)
  - *Short description:* **Display escalate to manufacturer for Infinity (HHD) service offerings when Priority = 1-Critical**
  - *Add Filter condition*: **Priority | is | 1-Critical AND**
  - *Add Filter condition*: **Service Offering | is | Infinity (HHD)**
  - *On load*: **checked**
  - *Reverse if false*: **checked**
- [ ] **Save** the **UI Policy** record.

(*Section 4: Create UI Policy Actions*)

- [ ] Create a **new UI Policy Action**:
  - *Field name*: **Escalate to manufacturer**
  - *Mandatory*: **Leave alone**
  - *Visible*: **True**
- [ ] Create another new **UI Policy Action**:
  - *Field name*: **Manufacturer contact**
  - *Mandatory*: **Leave alone**
  - *Visible*: **True**

(*Section 5: Verify UI Policy and Actions*)

- [ ] Verify by creating a new Incident with *Impact* and *Urgency* at **High** to update *Priority* to **1-Critical**. Both fields should display per the **UI Policy** that was created.
- [ ] Verify also that by changing *Impact* to **Medium** to remove the **Priority 1** condition you should see both fields disappear from the form.

## ➔ Business Rules

### 💡**Knowledge** [*SNAF Module 8.1 - UI Policies and Business Rules]*:

- [ ] Know what a **Business Rule** and the **4** settings for **When** it can be executed
  - **Business Rule**: Configured to run when a record is displayed, inserted, updated, deleted or when a table is queried
  - 1. *Table* to run against
  - 2. Timing of execution (*When to run*)
  - 3. Conditions to evaluate (*IWhen to run*)
  - 4. *Actions* to run
- [ ] Know what **Catalog UI Policies** can do and their **3** options for **When to Run**
  - **Catalog UI Policy**: type of UI Policy applied to a Catalog Item form
  - 1. onLoad()
  - 2. onChange()
  - 3. onSubmit()

### 📝**Practice Tasks** [*SNAF Lab 8.1 - Create a UI Policy and Business Rules*]

(*Section 6: Create Business Rules*)

- [ ] Configure a new **Business Rule** on any incident record:
  - *Name*: **Escalate to Manufacturer reminder**
  - *Table*: **Incident [incident]**
  - *Add Filter condition*: **Service Offering | is | Infinity (HHD) AND**
  - *Add Filter condition*: **Escalate to Manufacturer | is | True**
  - *Insert*: **checked**
  - *Update*: **checked**
- [ ] Under the *Actions* tab click **Add Message** and type: "**Reminder: to escalate to manufacturer, send mail to Manufacturer contact.**" Click **Submit**.
- [ ] Create another new **Business Rule**:
  - *Name*: **Populate manufacturer contact name**
  - *Table*: **Incident [incident]**
  - *Add Filter condition*: **Service Offering | is | Infinity (HHD) AND**

- *Add Filter condition*: **Priority | is | 1-Critical**
- *Insert*: **checked**
- *Update*: **checked**

- [ ] Under the *Actions* tab, set:
  - *Set Field Values*: **Manufacturer contact | Same as | Service Offering.Manufacturer.Contact** and click **Submit**

(*Section 7: Verify Business Rules work*)

- [ ] Verify by creating a new Incident with **Priority 1** and ensure both fields appear. **Save**.
- [ ] Confirm the **Manufacturer contact** displays and set it to **Test Contact**. Select the **Escalate to Manufacturer** checkbox and click **Save**.
- [ ] Verify the Reminder message displays.

## ➔ System Update Sets - Creating and Applying

- **Creating**

### 💡**Knowledge** [*SNAF Module 8.2 - Migration and Integration]*:

(*Section 1: Review the Default Global Update Set*)

- [ ] Know what an **Update Set** is and what **3** pieces of information it contains
  - **Update Set**: group of configuration changes that can be moved from one instance to another
  - 1. set of record details
  - 2. list of configuration changes
  - 3. state that determines whether or not the Update Set can be retrieved or applied to another instance
- [ ] Know what system table stores all the tracked table changes
  - **sys_update_xml**
- [ ] Know how to navigate to view existing **Update Sets**
  - **All > System Update Sets > Local Update Sets**
- [ ] Know that **Update Sets** can be batched and why you would avoid dealing with multiple update sets (**2** reasons)
  - You can create **batch update sets** which enable you to group update sets together so you can preview and commit them in bulk
  - Reason 1. You may commit the update sets in the wrong order
  - Reason 2. You may leave out an update set inadvertently
- [ ] Know **what is** and **what is not** captured within an **Update Set**
  - **Process Records** are captured, **Data** is not
- [ ] Know what function can be used if you need to export data manually
  - **Export XML** function
- [ ] Know how to create a new **Update Set** and select as current
  - **All > System Update Sets > Local Update Sets** and click **New**
  - Click **Submit and Make Current** once record is filled out
- [ ] Know how to mark an **Update Set** as complete and the risks in reopening a completed set
  - Change *State* to **Complete**
  - You should never move a **Complete** Update Set back to **In Progress**. Instead create a new **Update Set** to capture the remaining changes and commit them as a batch

### 📝**Practice Tasks** [*SNAF Lab 8.2 - Create System Update Sets*]

- [ ] Ensure **Scope** is set to **Global** and **Update Set** is **Default [Global]** before continuing
- [ ] Open the list of **Local Update Sets** (either by shortcut from globe, or from All menu)
- [ ] Locate and open the **Default** Update Set within the **Global** Application
- [ ] Scroll down to the *Customer Updates* tab and see what was captured from class labs

(*Section 2: Create an Update Set*)

- [ ] Create a **New** Local Update Set as follows:
  - *Name*: **Lab Update Set**
  - *State*: **[leave as In Progress]**
  - *Description*: **Catalog item form, add Updated and Updated by fields**
- [ ] Click **Submit and Make Current**, verify correct set shows in **Scope** (globe menu)

(*Section 3: Modify the Catalog Item Form*)

- [ ] Bring up the **Infinity HHD Catalog Item** record and use **Form Design** to add:

- *Updated* after *Active*
- *Updated by* after *Updated*

- [ ] **Save** and reload the form. Verify additions were made correctly.
- [ ] Go back to the **Lab Update Set** and verify there are updates with *Type* **Form Layout** and *Target name* **Catalog Item**.

- **Applying**

💡**Knowledge** [*SNAF Module 8.3 - Applying an Update Set]*:

- [ ] Know how to navigate to and apply an **Update Set**
  - **All > System Update Sets > Retrieved Update Sets**
  - Click **Import Update Set from XML** under *Related Links* and upload the file
- [ ] Know the typical **3-step** process of retrieving an **Update Set**
  - 1. Retrieve
  - 2. Preview
  - 3. Commit
- [ ] Know that during **Preview** what fields should match on platform records
  - **sys_id** fields
- [ ] Know how to **Update Sources** for an Update Set and test Connection for sub-production instances
  - **All > System Update Sets > Update Sources**
  - Once record is filled out click the **Test Connection** button
- [ ] Know how to **Commit** an Update Set
  - **All > System Update Sets > Retrieved Update Sets**
  - Select the update set, then click the **Commit Update Set** button

📝**Practice Tasks** [*SNAF Lab 8.3 - Migrate Changes to Test Instance*]

(*Section 1: Mark Update Set Complete and Export Update Set*)

- [ ] Open the record for **Lab Update Set** and verify changes from Lab 8.2 are there
- [ ] Change the **Update Set** *State* from **In Progress** to **Complete** and **Save** the record
- [ ] In the *Related Links* section, select **Export to XML**

(*Section 2: Prepare a PDI*)

(*Section 3: Verify Current Mark Update Set Complete and Export Update Set*)

- [ ] On your PDI, Simply verify that a Catalog Item with a *Category* of **Hardware** does not show the *Updated* or *Updated by* fields on the form

(*Section 4: Create and Retrieve an Update Set*)

- [ ] On your PDI instance, navigate to **All > System Update Sets > Retrieved Update Sets.**
- [ ] Scroll to the *Related Links* section and select **Import Update Set from XML**
- [ ] Locate the exported XML file from earlier and **Upload** it, confirm *State* is **Loaded**.
- [ ] Open the **Update Set** and click **Preview Update Set**
- [ ] Review and click **Commit Update Set** to commit. Close the dialog box once 100%.

(*Section 5: Locate Retrieved Update Sets*)

- [ ] Navigate to **All > System Update Sets > Retrieved Update Sets** and verify that the **Lab Update Set** has a *State* of **Committed**.
- [ ] Lastly, go check anyCatalog Item with a *Category* of **Hardware** and verify the *Updated* and *Updated by* fields are now visible on your PDI

➔ ATF / App Engine Studio

💡**Knowledge** [*SNAF Module 8 - Package Enhancements for Testing]*:

- [ ] Know what the **A**utomated **T**est **F**ramework and the benefits it provides
  - Create and run automated tests on your instance after modifying it
  - **Benefits**:
    - Reduce upgrade and development time by replacing manual testing
    - Design tests once and reuse them in different contexts
    - Keep test instances clean by rolling back test data and changes made after each test run
    - Create test suites to organize and run tests in batches
    - Schedule test suite runs
    - Enable non-technical test designers to create tests of standard ServiceNow Platform functionality

        - Reduce test design time by copying quick start tests and test suites
        - Create custom test steps to expand test coverage
- [ ] Understand what **App Engine Studio** is used for and the benefits it provides
    - Used for low code development, automating business processes and solving business problems.
        - Allows you to delegate development work that was once assigned to administrators, with little to no training
        - Leverage development tools and build apps quickly using templates for pre-built solutions
- [ ] Know the concept of **Designate Developers**
    - Non-administrative users and groups which are assigned one or more permissions to develop applications