



# REPORTING

21 novembre 2025

## 1. ITIL - ITSM

### 1) ITSM

Incident, SLA, problem, change, CMDB, service catalog, knowledge, service portal, connect, visual task board

### 2) Modules

- ITSM : Gestion des services IT (incidents, demandes, changements, problèmes).
- ITOM : Surveillance et gestion opérationnelle de l'infrastructure IT. -> Discovery, logs, service mapping infrastructure, cloud management, orchestration
- CSM : Expérience et gestion clients -> SLA, contrats, portail clients
- HRSD : Employés et RH Portail RH et gestion des services aux employés.
- GRC : Gestion des risques, conformité et audit. -> gestion risques, conformité réglementaire, suivi exigences légales, suivi actions correctrices, planification audit, gestion risque tiers
- ITBM : Gestion de portefeuilles et de projets IT. -> gestion portefeuilles, gestion demandes, projets, financière, tableaux de bord stratégiques
- SecOps (Security Operations) : Réponse aux incidents de sécurité et gestion des vulnérabilités. -> détection/résolution incidents, scans, réponses automatiques, conformité sécurité, tableaux bord sécurité
- App Engine : Création apps low code. -> applications low code service now, studio développement, Db, intégrations, déploiement, marketplace
- Now Platform : Infrastructure. Fondation technique commune à tous les modules. -> Db, moteur wkf, UI interfaces, sécurité, accès, intégration (REST, MID Server, SOAP), API, SDK

### 3) Fonctionnalités

#### - Incident

- > sources: email, portail SN, monitoring tool, intégration systèmes 1/3
- > lien business service/organisation, Configuration item, assign to group/triage
- > lien problème, change, tâches
- > classification impact, urgence
- > communication, notifications

#### - Problèmes

- > lien avec plusieurs incidents, impact global, tâches spécifique pour investiguer problème (analyse logs, contacter éditeur)
- > liens business service, configuration item
- > Workarounds à communiquer users en attendant résolution problème
- > lien change request: & X cause identifiée, un changement peut être nécessaire
- > Root cause
- > lien knowledge base pour documenter solution

## Différence clé avec Incident Management

Incident Management	Problem Management
Restaurer le service rapidement	Trouver la cause permanente
Solution temporaire	Solution définitive
Focus sur l'utilisateur	Focus sur l'infrastructure

### - Changements

- Ex: montée version (release) logiciel, patch sécurité, ajout/suppression serveur, changer règles Firewall, MFA, donner droits accès
- Se fait sur un CI: serveur applicatif ou autre
- On déploie un change de façon anticipée contrairement à 1 incident
- Type
  - Standard: préapprouvé (eg: création compte utilisateur)
  - Normal: approbation CAB requise, eg montée version application
  - Emergency: système critique attaqué besoin patch sécurité
- documentationImpact changement, risque
- Record change\_table
- Gestion conflit: vérifier si changement entre en conflit avec maintenance window, ou autre changement sur le même CI
- Workflow selon type de change: soumission, évaluation, documentation fenêtre maintenance/rollbackplan/risques/conflicts, implementation (inc. tests), clôture

### - Configuration Item/CMDB

- Pourquoi CMDB crucial: identifier sur quoi portent incidents, gérer cycle de vie des actifs, service mapping (CIs par business), compliance
  - Ex: serveurs, desktops, software, router, switches, contrats, documents
  - cmdb\_CI table (cmdb\_ci\_server, cmdb\_ci\_computer, cmdb\_ci\_appl, cmdb\_ci\_network)
    - 1 serveur aura des champs "CPU", "RAM", "OS" alors qu'une licence logicielle aura "Date d'expiration", "Nombre d'utilisateurs"
    - relationships entre CIs: Le serveur SRV001 héberge l'application SAP
    - dependency view map: visualisation graphique dépendances. Si le switch réseau tombe, on voit immédiatement quels serveurs et applications sont affectés
    - import: discovery, manual import. Discovery scanne réseau pour peupler CMDB.
    - concepts:
      1. Éléments de Configuration (CI): inventaire des actifs (serveurs, applications, réseaux, contrats), classés par type (VM), avec leurs attributs (srv- prod-dg05/réseau IP...) et cycle de vie (en production, retiré).
      2. Relations et Dépendances: Cartographie des liens entre CI pour analyser l'impact des pannes ou des changements sur les services métiers.
      3. Gouvernance et Sécurité:
        - Gestion des droits d'accès, traçabilité des modifications (audit trail) et politiques de sécurité pour protéger l'intégrité des données.
        - ACL : Contrôle par table/enregistrement/champ. Rôles : Reader, Editor, Manager, Admin
      4. Peuplement et Découverte: alimentation CMDB via formulaires, intégration API, imports CSV/xls, discovery, connecteurs cloud
      5. Qualité et Maintenance: réconciliation des doublons, indicateurs de santé (taux doublons, taux complétion), mises à jour régulières (équipe IT, discovery, réconciliation doublons)
      6. Usages Opérationnels: CMDB est 1 support à la gestion des incidents, planification changements, des actifs et au catalogue de services.
      7. Modèle de Données et Architecture: Schéma technique (tables, identifiants uniques) et mécanismes de fédération avec d'autres CMDB.
      8. Cost Model et Allocation des Coûts: modélisation financière des CI : attribution des coûts (achat, maintenance, énergie) aux services ou départements. Objectif : Optimiser les dépenses IT et responsabiliser les métiers.
        - Chargeback : Facturation interne aux utilisateurs (ex. : un département paie pour l'usage d'un serveur).
        - Showback : Transparence sur les coûts sans facturation inclus dans baseline (ex. : rapports pour sensibiliser à la consommation).
      9. Intégrations et Écosystème: liens avec outils ITSM, monitoring, cloud pour synchroniser les données et enrichir l'analyse financière.
      10. Reporting et Analyse: Tableaux de bord pour mesurer la qualité des données, l'impact des changements, et les coûts par service/actif.

```

CI : Serveur Web PRD-WEB-01
|--- Table : cmdb_ci_server
|--- Champs : IP=192.168.1.10, OS=Linux, RAM=16GB
|--- Relations :
|   |--- Héberge → Application "Site Internet Corporate"
|   |--- Connecté à → Switch "SW-CORE-01"
|--- Découvert par : Discovery Schedule "Nightly Scan"

```

### - Service catalog

-Offre de services: infrastructure (hardware, software, sécurité, réseau), support, coûts (CB/baseline), accès (business),

-Suivi requêtes, notifications

-Tables

sc\_cat\_item (surface pro)

sc\_request (requête/commande user)

sc\_task (validation manager > provisionnement > livraison)

-Request item: une même demande peut générer 3 items -> 1 ordinateur portable, accès wifi ,licence 365

-Workflow:

-Utilisateur → Parcourt sc\_cat\_item → Commande → Création sc\_request → Déclenchement sc\_tasks → Suivi progression → Livraison

-Use cases: Pour l'utilisateur :"Je vais dans le Service Catalog → Hardware → Je choisis 'Ordinateur Portable Standard' → Je remplis le formulaire → Je vois où en est ma demande"

Pour l'IT :"La demande crée automatiquement des tâches pour l'approbation budgétaire du manager, puis la configuration, puis la livraison logistique"

-Si commande > 1000€ → Approbation financière obligatoire

-Spécifique à chaque service

-User criteria -> eg seuls les managers peuvent demander des licences photoshop

### -Knowledge base:

-Categories

-Articles per category

-Approval

-Feedback, lifetime, population

### - SLA

-> Paramétrable par département/organisation

-> Pause si attente côté client

-> Exemples:

Incident: temps de résolution, temps de première réponse

Problème: délai d'analyse racine

Change: délai d'approbation, délai de livraison

Service Catalog SLA: temps de traitement de la requête

Asset Management SLA: délai de provisionnement

## 4) Technique

### -Intégration third-party

-LLM OpenAI, IA automatisation (N8N), Cloud AWS

-N8N: dans N8N connecteur SN, N8N appelle API REST Service Now (connection OAuth 2.0, SN peut déclencher N8N via webhooks)

### -Intelligence Artificielle

-Technologie Now Assist : Assistant IA intégré partout, virtual agents (chatbots intelligents)

-Use cases IA

-Classification automatique des incidents avec >95% de précision

-Routing intelligent vers les bons groupes de support

-Suggestions en temps réel de Solutions, articles Knowledge Base

-Détection comportements systèmes anormaux

-LLM: trouve mes incidents en cours, recherche globale cross-tables

- Virtual Agent Avancé: 40-60% des incidents résolus sans agent humain. Réponses automatiques aux clients
- Escalation fluide vers agent humain avec contexte complet. Context awareness : Comprend l'historique utilisateur
- Prédiction des pannes : Alertes avant que l'incident ne survienne
- Risk assessment : Évaluation automatique des risques associés aux changement
- Demand forecasting : Prévision des volumes de tickets
- SLA breach prediction : Alerte avant dépassement SLA
- Génération low-code

#### **-Développement**

API

#### **5) Autres**

##### **-Service Portal**

- For end users, UI/UX
- bootstrap, angularjs
- responsive

##### **-Visual Tasks Board**

##### **-Connect**

-Chat pour accélérer support clients, discussions entre équipes, liens incidents/demandes/changements, routing vers bonnes équipes, multi-queue



## **2. ITIL - FOUR DIMENSIONS**

### **1. Organizations & People: Culture, Skills, Structures.**

This refers to the human element of the organization.

- Culture: The shared values, behaviors, and mission.
- Skills: The competencies and capabilities of the workforce.
- Structures: The organizational hierarchy, teams, and reporting lines.

### **2. Information & Technology: Data, Knowledge, Applications, Infrastructure.**

This covers the digital and information assets that enable the business.

- Data: The raw facts and figures.
- Knowledge: The insights and understanding derived from data.
- Applications: The software and systems used (e.g., CRM, ERP).
- Infrastructure: The underlying technology (networks, servers, cloud platforms).

### **3. Partners & Suppliers: Contracts, Relationships, Integration.**

This defines how the organization works with external entities.

- Contracts: The formal agreements and service level agreements (SLAs).
- Relationships: The nature of the partnerships (strategic, transactional).
- Integration: How seamlessly external partners are connected to internal processes and systems.

### **4. Value Streams & Processes: How work is accomplished.**

- This describes the core activities that deliver value to the customer.
- Value Streams: The end-to-end sequences of activities that create a product or service.
- Processes: The specific, defined steps for completing a task or function.



## **3. SN - SOURCES**

- Créer un compte communautaire ServiceNow : community.servicenow.com
- Documentation produit : docs.servicenow.com
- Tutoriels vidéo : Chaîne YouTube ServiceNow

-Dévs: [https://www.udemy.com/course/servicenow-201-development/?-couponCode=25BBPMXINACTIVE\(dev\)](https://www.udemy.com/course/servicenow-201-development/?-couponCode=25BBPMXINACTIVE(dev))

-Others

<https://youtu.be/6-VCcmP0WkA> (dev)

<https://www.youtube.com/watch?v=QxU-JrfA824&feature=youtu.be> (non dev)

<https://www.youtube.com/watch?v=o9TubFmirwc> (CMDB)

<https://www.youtube.com/watch?v=D5hzOJMI6u8> (CMDB)

<https://www.youtube.com/watch?v=CgY8iJZ-DhA&pp=0gcJCQgKAYcqIYzv> (CMDB)

<https://www.youtube.com/watch?v=0i-6VOHa2dg> (CMDB)



## 4. 7) SN - ITSM CORE APPLICATIONS

- 1.Service Desk
- 2.Incident Management
- 3.Problem Management
- 4.Change Enablement (Change Management)
- 5.Service Request Management
- 6.Service Level Management
- 7.Service Configuration Management
- 8.IT Asset Management
- 9.Monitoring and Event Management
- 10.Release Management
- 11.Service Catalog Management
- 12.Availability Management
- 13.Capacity and Performance Management
- 14.Service Continuity Management
- 15.Deployment Management
- 16.InfoSec Management (Information Security Management)
- 17.Supplier Management

### 1. Service Desk

Purpose: To capture demand for incident resolution and service requests. It is the single point of contact (SPOC) between the service provider and users.

Keywords: Single Point of Contact (SPOC), Incident Logging & Categorization, User Communication & Notification, Request Fulfillment, Call Center, Contact Center, Help Desk, First-Line Support / Level 1 Support, Communication Channel (Phone, Email, Chat, Self-Service), Service Portal / Self-Service Portal, Business Hours, Automated Call Distribution (ACD), Workforce Management (WFM)

Key Activities:

- Logging, categorizing, and prioritizing incidents and service requests.
- Providing first-line investigation and resolution (e.g., password resets).
- Keeping users informed about the status of their requests.
- Communicating service outages and planned changes to users.

Examples:

-A user calls to report that they cannot access their email. The service desk agent logs an incident, provides initial troubleshooting, and escalates it to the messaging team when they cannot resolve it.

-The service desk sends a mass notification to all users informing them of a scheduled network maintenance window this weekend, acting as the central communication point.

### 2. Incident Management

Purpose: To reduce the negative impact of incidents by restoring normal service operation as quickly as possible.

Keywords:Incident, Restoration of Service, Service Disruption, Impairment, Service Level Agreement (SLA), Major Incident, Incident Categorization & Prioritization, Escalation (Functional, Hierarchical), Incident Model / Process Flow, Service Downtime, Impact & Urgency, Initial Diagnosis, Resolution, Incident Closure

**Key Activities:**

- Identifying, logging, and categorizing incidents based on urgency and impact.
- Prioritizing incidents and escalating them to appropriate support teams (e.g., second-line, third-line).
- Investigating and diagnosing the root cause to resolve the incident.
- Closing incidents and confirming resolution with the user.

**Examples:**

- A major outage occurs where the company's CRM system is down. The incident is prioritized as a P1 (highest priority), and all relevant support teams are mobilized to restore service immediately.
- A single user reports that their spreadsheet application is crashing. This is logged as a P4 (low priority) incident, and a support technician is assigned to resolve it within the agreed SLA.
- The monitoring tool automatically creates an incident because a server's CPU usage is at 100%. The incident management process ensures the alert is routed to the server support team for investigation.

**3. Problem Management**

Purpose: To reduce the likelihood and impact of incidents by identifying actual and potential causes of incidents and managing workarounds and known errors.

Keywords: Problem, Root Cause Analysis (RCA), Workaround, Known Error, Knowledge Base (KEDB), Proactive Problem Management, Reactive Problem Management, Ishikawa (Fishbone) Diagram, The 5 Whys, Permanent Solution, Problem Categorization & Prioritization, Error Control

**Key Activities:**

- Identifying and logging problems from recurring or major incidents.
- Performing root cause analysis (RCA) to find the underlying cause.
- Creating known error records in the Known Error Database (KEDB).
- Developing permanent fixes or workarounds to be implemented via Change Enablement.

**Examples:**

- The problem management team notices a trend of multiple incidents related to a specific network switch failing. They open a problem record, perform an RCA, and find a faulty hardware batch, leading to a replacement plan.

**4. Change Enablement (Change Management)**

Purpose: To maximize the number of successful service and product changes by ensuring risks are properly assessed, authorizing changes to proceed, and managing the change schedule.

Keywords: Change, Standard Change (Pre-authorized, Low Risk), Normal Change, Emergency Change, Change Advisory Board (CAB) / Emergency CAB (ECAB), Authorization, Change Schedule (Forward Schedule of Change - FSC), Change Model, Request for Change (RFC), Remediation Plan (Backout Plan), Change Implementation, Post-Implementation Review (PIR), Change Type (Standard, Normal, Emergency)

**Key Activities:**

- Receiving and logging change requests (Normal, Standard, Emergency).
- Assessing, evaluating, and authorizing changes via a Change Advisory Board (CAB).
- Scheduling and coordinating the implementation of changes.
- Reviewing and closing changes to ensure they were successful.

**Examples:**

- A normal change is submitted to upgrade a core database server. The CAB meets to review the implementation plan, back-out plan, and risk assessment before authorizing it for a scheduled maintenance window.
- A standard change is initiated to onboard a new employee, which follows a pre-approved, low-risk script to create user accounts and assign software licenses.
- An emergency change is rushed through a streamlined process to apply a critical security patch to a production system in response to a newly discovered vulnerability.

**5. Service Request Management**

Purpose: To support the agreed quality of a service by handling all pre-defined, user-initiated service requests in an effective and user-friendly manner.

Keywords: Service Request, Standard Change (often used to fulfill requests), Pre-defined, Pre-approved, User-Initiated, Service Catalog Item, Request Model, Fulfillment, Self-Service, Approval Workflow

**Key Activities:**

- Defining and cataloging standard service requests (e.g., access to an application, new equipment).
- Providing a user-friendly portal for submitting requests.
- Fulfilling requests, often using pre-approved procedures and catalogs.

**Examples:**

- An employee requests a new database, server

**6. Service Level Management**

Purpose: To set clear business-based targets for service performance, and to ensure that delivery of services is properly assessed, monitored, and managed against these targets.

Keywords: SLA (Service Level Agreement), OLA (Operational Level Agreement), Underpinning Contract (UC), Service Review Meeting, Service Level Target (SLT), Service Level Requirement (SLR), Key Performance Indicator (KPI), Service Quality, Service Improvement Plan (SIP), Monitoring & Reporting

**Key Activities:**

- Negotiating, agreeing upon, and documenting Service Level Agreements (SLAs) with customers.
- Monitoring and reporting on service performance against SLA targets.
- Conducting regular service reviews with customers and stakeholders.
- Managing underpinning contracts and Operational Level Agreements (OLAs).

**Examples:**

- An SLA with the Sales department states that the CRM system will have 99.9% uptime during business hours, and that priority 1 incidents will be resolved within 4 hours.
- An OLA is established with the network team, committing them to resolve any network-related issues within 1 hour to help the service desk meet its 4-hour SLA.
- During a quarterly service review, the IT manager presents a report to the business showing that uptime was 99.95% and discusses plans to improve resolution times for P2 incidents.

**7. Service Configuration Management**

Purpose: To ensure accurate and reliable information about the configuration of services, and the Configuration Items (CIs) that support them is available when and where it is needed.

Keywords: Configuration Item (CI), Configuration Model, CMDB (Configuration Management Database), Relationship (e.g., parent/child, connects-to, depends-on), Attribute, CI Lifecycle, Baseline, Snapshot, Verification & Audit, Definitive Media Library (DML)

**Key Activities:**

- Identifying, controlling, and accounting for Configuration Items (CIs).
- Maintaining information about CIs in a Configuration Management Database (CMDB).
- Verifying and auditing the CMDB data to ensure its accuracy.

**Examples:**

- A CI record for a server in the CMDB contains its model, serial number, IP address, and shows its relationships to the application it hosts and the business service that depends on it.
- When a change is made to upgrade the server's memory, the CMDB is updated to reflect the new configuration, maintaining an accurate record.
- An automated discovery tool scans the network and compares the live environment to the CMDB, flagging any discrepancies for reconciliation (e.g., an unregistered virtual machine).

**8. IT Asset Management**

Purpose: To plan and manage the full lifecycle of all IT assets to help the organization maximize value, control costs, manage risk, support decision-making, and meet regulatory and contractual requirements.

Keywords: Asset, Lifecycle (Plan, Acquire, Deploy, Manage, Retire), Cost, Value, Risk, Financial Management, Software Asset Management (SAM), Hardware Asset, Asset Register, Total Cost of Ownership (TCO), Depreciation, Compliance & Licensing

**Key Activities:**

- Tracking the financial, contractual, and inventory details of IT assets throughout their lifecycle (procurement to disposal).

**Examples:**

- Tracking the lifecycle of 500 laptops from purchase, through deployment and usage, to their eventual secure wiping and disposal or resale after 4 years.
- Managing software license counts and usage to ensure compliance during a vendor audit, avoiding hefty fines for non-compliance.
- Analyzing asset data to make a cost-saving decision to extend the refresh cycle for desktops from 3 to 4 years, as they remain fit for purpose.

**9. Monitoring and Event Management**

Purpose: To systematically observe services and service components, and to record and report selected changes of state identified as events.

Keywords: Event, Monitoring, Alert, Notification, Filtering, Correlation, Event Type (Informational, Warning, Exception), Performance Monitoring, Proactive Monitoring, System Log

**Key Activities:**

- Monitoring services and infrastructure components for changes of state (events).
- Categorizing events as Informational, Warning, or Exception.
- Automating responses to certain events to prevent incidents.
- Triggering incident records for exception events that indicate a service failure.

**Examples:**

- A monitoring tool generates an informational event that a scheduled backup job completed successfully.
- A warning event is generated when a database's free space drops below 20%. An automated script adds more space, preventing a service disruption.
- An exception event is created when a critical web server becomes unresponsive. This event automatically triggers the creation of a P1 incident in the service desk tool.

**10. Release Management**

Purpose: To make new and changed services and features available for use.

Keywords: Release, Deploy / Deployment, Build, Test, Early Life Support (ELS), Release Package, Release Policy, Deployment Management (often integrated), Release Unit, Rollout, Backout Plan

**Key Activities:**

- Planning the release schedule and release package.
- Designing, building, and testing a release in a controlled environment.
- Deploying the release into the live environment (production).

**Examples:**

- A major release of a new mobile banking app, which includes new features and is deployed to app stores for users to download.
- A minor release (or "hotfix") that deploys a small patch to the company's intranet to fix a specific bug, with minimal impact on users.
- An emergency release to deploy a security update across all company workstations using an automated deployment tool outside of the normal schedule.

**11. Service Catalog Management**

Purpose: To provide a single source of consistent information on all services and service offerings, and to ensure that it is available to the relevant audience.

Keywords: Service Catalog, Live Services, Service Pipeline, Service Portfolio, User-facing Services / Customer-facing Services, Supporting Services, Business Service, Technical Service, Service Definition, Service Offering

**Key Activities:**

- Developing and maintaining the Service Catalog with details on all live services and those available for deployment.
- Ensuring the catalog is accessible to users and customers.

**Examples:**

- The customer-facing service catalog lists business-level services like "Recruitment Management" or "Sales Lead Tracking" for business managers to view.
- The user-facing section of the catalog allows employees to request "Email and Collaboration," "HR Self-Service," or "Standard Workstation" services.
- The catalog is updated to include a new "Video Conferencing" service after it has been tested and is ready for deployment, complete with description and cost.

## 12. Availability Management

Purpose: To ensure that services deliver agreed levels of availability to meet the needs of customers and users.

Keywords: Availability, Reliability, Maintainability, Serviceability, Resilience, Vital Business Function (VBF), MTBF (Mean Time Between Failures), MTBSI (Mean Time Between Service Incidents), MTRS (Mean Time to Restore Service), Redundancy, Fault Tolerance

Key Activities:

- Monitoring and analyzing service availability, reliability, and maintainability.
- Identifying and mitigating risks to availability.
- Planning and designing infrastructure for resilience and redundancy.

Examples:

- Designing a web server farm with redundant servers behind a load balancer so that if one server fails, the others continue to handle traffic, maintaining availability.
- Analyzing the reliability of a key application by tracking the mean time between failures (MTBF) and working to extend it through software improvements.
- Implementing a high-availability (HA) cluster for a critical database to ensure it can automatically failover to a secondary node with minimal downtime.

## 13. Capacity and Performance Management

Purpose: To ensure that services achieve agreed and expected performance, satisfying current and future demand in a cost-effective way.

Keywords: Capacity, Performance, Demand, Utilization, Scaling (Horizontal, Vertical), Threshold, Workload Management, Performance Model, Capacity Plan, Bottleneck Analysis, Tuning

Key Activities:

- Monitoring performance and capacity of services and components.
- Analyzing utilization trends and forecasting future capacity requirements.
- Planning for and managing scaling to meet business demand.

Examples:

- Monitoring CPU and memory utilization on virtual hosts and forecasting that new hardware will be needed in 9 months to handle projected growth.
- Analyzing network throughput during peak business hours and upgrading the internet connection to prevent bottlenecks that slow down critical applications.
- Using auto-scaling in a cloud environment to automatically add more web servers during a high-traffic sales event, then scaling back down afterward to control costs.

## 14. Service Continuity Management

Purpose: To ensure that the availability and performance of a service are maintained at sufficient levels in case of a disaster.

Keywords: IT Service Continuity Management (ITSCM), Business Impact Analysis (BIA), Recovery Time Objective (RTO), Recovery Point Objective (RPO), Disaster Recovery, Business Continuity Plan (BCP), Risk Assessment, Continuity Strategy, Testing & Exercising

Key Activities:

- Performing a Business Impact Analysis (BIA).
- Developing and maintaining service continuity plans.
- Conducting regular tests of the plans and recovery procedures.

Examples:

- A BIA identifies the payroll system as critical, with a Recovery Time Objective (RTO) of 4 hours and a Recovery Point Objective (RPO) of 15 minutes after a disaster.
- A service continuity plan is created that details how to failover the entire data center to a secondary disaster recovery site.
- An annual test is conducted to restore the CRM system from backups at the DR site to validate the RTO and ensure the team is familiar with the procedure.

## 15. Deployment Management

Purpose: To move new or changed hardware, software, documentation, processes, or any other component to live environments.

Keywords: Deploy, Move, Add, Change (MAC), Build, Package, Release (often the input for deployment), Environment (Dev, Test, Staging, Prod), Automated Deployment, Physical Deployment, Virtual Deployment, Deployment Log

#### Key Activities:

- Planning and scheduling deployments.
- Executing the physical or automated movement of components.
- Verifying successful deployment and documenting the outcome.

#### Examples:

- Using an automated script to deploy a new version of an anti-virus client to all 5,000 company workstations.
- A technician physically installs and configures a new network switch in a branch office, following a standardized procedure.
- The deployment team rolls out a new version of the company's mobile app to a small pilot group of users first (canary release) before a full deployment.

## **16. InfoSec Management (Information Security Management)**

Purpose: To protect the information needed by the organization to conduct its business.

Keywords: Confidentiality, Integrity, Availability, CIA Triad, Security Policy, Risk Assessment, Information Security Management System (ISMS), Access Control, Threat, Vulnerability, Security Incident, Cybersecurity

#### Key Activities:

- Implementing and managing information security controls as defined by the organization's security policy.
- Managing access to services and data based on the principle of least privilege.
- Handling security incidents and conducting security audits.

#### Examples:

- Implementing a technical control that enforces multi-factor authentication (MFA) for all remote access to protect confidentiality.
- Following the principle of least privilege by ensuring users only have access to the specific data and systems required for their job role.
- Leading the response to a security incident involving a phishing attack, including containing the threat, eradicating the malware, and recovering the affected systems.

## **17. Supplier Management**

Purpose: To manage supplier relationships and their performance to ensure seamless and consistent quality service delivery.

Keywords: Supplier, Partner, Contract, SLA (in the context of a supplier), Performance Management, Relationship Management, Supplier Strategy, Supplier Categorization, Contract Lifecycle, Underpinning Contract (UC), Service Integration and Management (SIAM)

#### Key Activities:

- Establishing and managing supplier contracts and agreements (Underpinning Contracts).
- Monitoring and reviewing supplier performance.
- Managing supplier risks and fostering strong relationships.

#### Examples:

- Negotiating and managing an Underpinning Contract (UC) with a cloud hosting provider that includes SLAs matching the business's internal SLAs for uptime.
- Holding a quarterly performance review with a software vendor to discuss their performance against agreed metrics and any ongoing issues.
- Identifying a risk where a critical service relies on a single supplier and developing a strategy to find a secondary supplier to mitigate that risk.

## **5. 2 SN - TECH ARCHITECTURE**

Voir documentation technique SN

#### Principes Architecture

- Cloud (SaaS) : Modèle standard, pas de maintenance matérielle ou logicielle, mises à jour automatiques (2 à 3 fois par an).

Accès sécurisé depuis n'importe où. Scalabilité et haute disponibilité garanties.

-On-Premise : Disponible uniquement pour des cas très spécifiques (ex. : gouvernements, secteurs réglementés comme la défense).

#### -Haute Disponibilité et DR (Disaster Recovery)

ServiceNow garantit une disponibilité de 99,9% grâce à :

Réplication des données entre datacenters.

Basculement automatique en cas de panne.

-Plan de reprise d'activité (DR) : Sauvegardes régulières et tests de restauration.

-Instances ServiceNow

Production (PROD) : Environnement live pour les utilisateurs finaux.

Non-Production (DEV, TEST, UAT) : Environnements pour le développement, les tests, et la validation.

### **Composants Clés de l'Architecture ServiceNow**

**-Mid Server:** agent léger installé on-premise (dans le réseau de l'entreprise) ou dans un cloud privé VPC

Fait le pont entre ServiceNow (cloud) et les systèmes internes. Ex. : Active Directory, bases de données locales, outils de monitoring.

Fonctions clés :

Discovery : Scan du réseau pour identifier les actifs (serveurs, applications, etc.) et les synchroniser avec la CMDB.

Intégrations : Exécution de scripts ou de commandes locales (ex. : lancer un script PowerShell pour récupérer des logs).

Orchestration : Automatisation de tâches entre ServiceNow et des outils internes (ex. : redémarrer un serveur via SSH).

#### **-Base de Données (Now Platform Database)**

Stocke toutes les données de ServiceNow (incidents, CMDB, catalogues de services, etc.).

Utilise un modèle relationnel avec des tables personnalisables (ex. : incident, cmdb\_ci, task).

#### **CMDB (Configuration Management Database)**

Base de données centralisée pour gérer les Configuration Items (CI) (serveurs, applications, réseaux, etc.) et leurs relations.

Service Mapping : Cartographie des dépendances entre CI pour analyser l'impact des incidents.

#### **-Application Servers**

Hébergent la logique métier et les applications ServiceNow.

Exécutent les scripts (JavaScript côté serveur), les workflows, et les intégrations.

Fonctions clés :

-Business Rules : Scripts déclenchés par des événements (ex. : envoyer un email quand un incident est créé).

-UI Policies : Règles pour afficher/masquer des champs dans les formulaires.

-Scheduled Jobs : Tâches planifiées (ex. : nettoyage des données obsolètes).

#### **-Interface Utilisateur (UI)**

Service Portal : Portail personnalisable pour les utilisateurs finaux (HTML, CSS, AngularJS).

Now Platform UI : Interface administrative pour les admins/développeurs.

Mobile App : Application dédiée pour les techniciens sur le terrain.

#### **-IntegrationHub et Flow Designer**

IntegrationHub : Outil low-code pour connecter ServiceNow à des applications tierces (ex. : Slack, Jira, SAP) via des spokes (connecteurs pré-construits).

Flow Designer : Outil visuel pour automatiser des processus métiers (ex. : approbations, notifications).

Exemple : Créer un flow qui envoie une notification Slack quand un incident critique est ouvert.

#### **-Event Management et Monitoring**

Nagios, Splunk pour Event Management : Centralise les événements (alertes, logs) depuis des outils de monitoring

Alertes : Déclenche des incidents ou des actions automatiques en fonction des événements.

Exemple: Une alerte CPU élevée dans un serveur déclenche automatiquement un incident dans ServiceNow.

#### **-Security et Authentication**

SSO (Single Sign-On) : Intégration avec des fournisseurs comme Okta, Azure AD.

RBAC (Role-Based Access Control) : Gestion fine des permissions (ex. : un technicien ne voit que les incidents de son équipe).

Encryption : Chiffrement des données en transit et au repos.

#### **-ServiceNow Store et Applications Communautaires**

Marketplace d'applications et de plugins développés par ServiceNow ou la communauté.

Exemples: Applications pour la gestion des vulnérabilités, l'ITAM avancé, ou des connecteurs spécifiques (ex. : Zoom, Teams).

### -Performance Analytics et Reporting

Outil intégré pour créer des tableaux de bord et des rapports personnalisés (ex. : temps moyen de résolution des incidents, satisfaction utilisateur).

KPIs : Suivi des indicateurs clés (ex. : MTTR, nombre d'incidents par service).

Export : Génération de rapports PDF/Excel pour les revues de service.

### -Now Platform et Extensibilité

Low-Code/No-Code : Outils comme App Engine Studio pour créer des applications sans coder.

Pro-Code : Possibilité d'écrire du JavaScript avancé pour des besoins complexes.

### -API et Web Services

REST API : Pour interagir avec ServiceNow depuis des applications externes.

SOAP API : Pour les intégrations legacy.

GraphQL : Pour des requêtes complexes et optimisées.

### -Automatisation avec RPA (Robotic Process Automation)

Intégration avec des outils RPA (ex. : UiPath, Automation Anywhere) : Automatiser des tâches répétitives (ex. : création de tickets depuis des emails, mise à jour de la CMDB depuis des fichiers Excel).



## 6. 3) SN - LISTS AND FORMS

### 1. Records (Enregistrements)

Ce qu'on peut faire : Créer, modifier, supprimer ou consulter une entité unique (ex. : un incident, un utilisateur, un serveur) dans une table ServiceNow.

### 2. Lists (Listes)

-Ce qu'on peut faire : Visualiser, trier, filtrer et manipuler plusieurs records simultanément pour une gestion rapide et efficace.

-Fonctionnalités :

Filtres et tri : Affiner et organiser les données (ex. : incidents par priorité).

Multi-sélection : Appliquer des actions en masse (ex. : assigner 10 tickets à un technicien).

Cellules interactives : Modifier directement une valeur dans la liste (ex. : changer un statut).

Pagination : Naviguer entre les pages pour les grandes listes.

Export : Exporter les données vers Excel ou PDF.

### 3. Forms (Formulaires)

-Ce qu'on peut faire : Afficher, éditer ou créer un record avec tous ses détails, pour une gestion approfondie et personnalisée.

-Fonctionnalités :

Champs détaillés : Accéder à toutes les informations d'un record (ex. : historique complet d'un incident).

Édition avancée : Modifier des champs complexes (ex. : zones de texte riche, listes dynamiques).

Onglets et sections : Organiser les champs pour une meilleure lisibilité (ex. : onglets Informations, Historique).

Champs liés : Voir les records associés (ex. : tâches liées à un incident).

Personnalisation : Adapter le formulaire à son rôle (ex. : masquer des champs inutiles).

### 4. Filtres et Recherche

-Ce qu'on peut faire : Trouver rapidement des records spécifiques grâce à des outils de recherche puissants et flexibles.

-Fonctionnalités :

Barres de recherche : Rechercher par mots-clés (ex. : "serveur down").

Wildcards : Utiliser des caractères génériques pour des recherches partielles (ex. : \*down\*).

Condition Builder : Créer des filtres complexes sans SQL (ex. : Priorité = 1 ET Catégorie = Réseau).

Accès aux champs liés : Filtrer en utilisant des champs de tables liées (ex. : incidents pour un utilisateur spécifique).

Sauvegarde de filtres : Enregistrer et partager des filtres réutilisables (ex. : "Mes incidents en retard").

### 5. Breadcrumbs (Fil d'Ariane)

-Ce qu'on peut faire : Naviguer facilement entre les vues et comprendre le contexte de sa recherche ou de ses filtres.

-Fonctionnalités :

Navigation dynamique : Retracer son parcours (ex. : Liste > Filtre > Record).

Suppression de filtres : Retirer un filtre en un clic (ex. : cliquer sur > Incidents P1 pour revenir à la liste complète).

Copier la requête : Partager un lien vers une vue filtrée (ex. : envoyer l'URL des "Incidents ouverts pour le service RH").

## 6. Context Menus (Menus Contextuels)

-Ce qu'on peut faire : Accéder rapidement à des actions courantes via un clic droit sur un record, dans une liste ou un formulaire.

-Fonctionnalités :

Menus dynamiques : Actions adaptées au contexte (ex. : Fermer, Escalader, Envoyer un email).

Personnalisation : Ajouter des actions spécifiques par rôle (ex. : option Redémarrer le serveur pour les admins).

## 7. Personnalisation des Lists & Forms

-Ce qu'on peut faire : Adapter les listes et formulaires à ses besoins pour une expérience utilisateur optimisée.

-Fonctionnalités :

Ajout/Retrait de colonnes : Choisir quels champs afficher dans une liste (ex. : ajouter Dernière mise à jour).

Réorganisation : Déplacer les colonnes ou champs pour une meilleure ergonomie.

Personnalisation utilisateur : Masquer des champs inutiles dans un formulaire (ex. : Coût pour un technicien niveau 1).

Configuration système : Définir des affichages par défaut pour tous les utilisateurs (ex. : colonnes obligatoires pour une équipe).



## 7. 2) SN - TECH LANGUAGES PROGRAMMATION

### Langages

JavaScript (cœur du développement ServiceNow)

-Scripts côté serveur GlideRecord, GlideAjax, GlideSystem et côté client.

-Glide API: Ensemble d'objets JavaScript propres à ServiceNow

-HTML/CSS/Angular JS: Pour personnaliser les interfaces (Service Portal, formulaires).

### Use cases

-Scripts de migration de données

Techno : GlideRecord + Transform Maps + CSV import

-Automatisation des tests de qualité des données

Valeur : Maintenance proactive CMDB

Techno : GlideRecord + Scheduled Jobs + Reports

-Détection et correction automatique des doublons

Techno : GlideRecord + Business Rules + JavaScript

-Automatisation avancée des workflows métiers. Ex: approbation dépenses

Techno : GlideRecord + JavaScript serveur + Workflow Editor

-Intégration avec systèmes externes (API REST). Ex: système HR pour garder poste salarié à jour.

Techno : REST API + GlideRecord + Scripted REST APIs

-Développement d'applications métiers sur mesure

Techno : App Engine + GlideRecord + REST API + UI Builder

-Tableaux de bord dynamiques avec données externes

Techno : REST API + Widgets + HTML/CSS/JavaScript

-Portail client/externe ticketing personnalisé avec AngularJS

Techno : AngularJS + Service Portal + CSS/HTML + API tierces

-Rappels et notifications intelligentes

Valeur : Respect SLA amélioré, réduction retards

Techno : GlideSystem + Email Notifications + Slack/Teams API

-Personnalisation des formulaires avec logique conditionnelle

Techno : Client Scripts + UI Policies + HTML/CSS



## 8. 3) SN - LISTS AND FORMS - CAS D'USAGE

### 1. Gestion des Incidents par le Service Desk

Scénario : Un technicien veut trier et prioriser les incidents ouverts pour son équipe.

Ce qu'il fait :

- Liste des incidents : Filtre les tickets avec Statut = Ouvert et Groupe = Service Desk Paris.
  - Tri par priorité : Affiche les incidents P1 en haut de la liste.
  - Multi-sélection : Assigne 5 incidents P2 à un collègue en une seule action.
  - Personnalisation : Masque la colonne Coût (inutile pour le niveau 1) et ajoute Dernière mise à jour.
- Valeur ajoutée :
- Gain de temps : Le technicien traite les incidents critiques en premier, sans perdre de temps à naviguer.
  - Collaboration : Les incidents sont rapidement assignés, réduisant les retards.

### 2. Suivi des Demandes de Service (Service Catalog)

Scénario : Un employé demande un nouvel accès logiciel via le portail self-service.

Ce qu'il fait :

- Formulaire de demande : Remplit un formulaire préconfiguré avec des champs obligatoires (Logiciel, Justification).
- Workflow automatique : La demande est routée vers le responsable pour approbation.
- Notification : L'employé reçoit un email avec le statut (Approuvé/Rejeté).

Valeur ajoutée :

- Automatisation : Plus besoin d'emails ou d'appels pour suivre la demande.
- Traçabilité : Toutes les étapes sont enregistrées dans le système.

### 3. Mise à Jour de la CMDB (Gestion des Actifs)

Scénario : Un administrateur veut mettre à jour les informations d'un serveur dans la CMDB.

Ce qu'il fait :

- Recherche dans la CMDB : Utilise le Condition Builder pour trouver le serveur par Nom ou IP.
- Formulaire détaillé : Modifie les attributs (Modèle, Responsable, Date de maintenance).
- Champs liés : Vérifie les applications dépendantes depuis l'onglet Relations.
- Audit trail : ServiceNow enregistre automatiquement qui a fait la mise à jour et quand.

Valeur ajoutée :

- Précision : La CMDB reste à jour, évitant les erreurs de gestion des actifs.
- Impact analysis : Savoir quelles applications seront affectées en cas de panne du serveur.

### 4. Gestion des Changements (Change Management)

Scénario : Un responsable IT veut approuver un changement planifié (ex. : mise à jour d'un serveur).

Ce qu'il fait :

- Liste des changements : Filtre les demandes avec Statut = En attente d'approbation.
- Formulaire de changement : Consulte les détails (Risques, Plan de retour arrière, Fenêtre de maintenance).
- Approbation en un clic : Utilise le menu contextuel pour Approuver ou Rejeter.
- Notification automatique : Le demandeur et l'équipe technique sont notifiés.

Valeur ajoutée :

- Contrôle : Les changements sont validés selon les processus ITIL, réduisant les risques.
- Transparence : Tout le monde est informé des décisions.

### 5. Résolution des Problèmes Récurrents (Problem Management)

Scénario : Un problème récurrent (ex. : plantage d'une application) doit être documenté et résolu.

Ce qu'il fait :

- Création d'un problème : Lie plusieurs incidents similaires via le champ Problème associé.
- Analyse RCA (Root Cause Analysis) : Utilise le formulaire pour documenter la cause racine et la solution.
- Base de connaissances : Enregistre la solution dans la Known Error Database (KEDB) pour référence future.

Valeur ajoutée :

- Prévention : Évite la récurrence du problème en documentant la solution.
- Efficacité : Les techniciens trouvent rapidement des solutions existantes.

### 6. Gestion des Utilisateurs (User Administration)

Scénario : Un administrateur doit mettre à jour les informations d'un employé (ex. : changement de département).

Ce qu'il fait :

- Recherche utilisateur : Utilise la barre de recherche avec Nom ou Email.
- Formulaire utilisateur : Modifie les champs Département, Rôle, et Accès.
- Champs liés : Vérifie les groupes et rôles associés pour ajuster les permissions.

Valeur ajoutée :

- Sécurité : Les accès sont mis à jour en temps réel, évitant les risques de droits obsolètes.
- Conformité : Respect des politiques RH et IT.

## 7. Automatisation des Rappels (SLA Management)

Scénario : Un manager veut s'assurer que les SLAs sont respectés pour les incidents critiques.

Ce qu'il fait :

- Filtre des incidents : Affiche les tickets avec Priorité = 1 et Temps restant < 2h.
- Condition Builder : Sauvegarde le filtre Incidents P1 en retard.
- Notification automatique : Une Business Rule envoie un email aux techniciens si le SLA est violé.

Valeur ajoutée :

- Respect des SLAs : Réduction des pénalités et amélioration de la satisfaction client.
- Proactivité : Les équipes sont alertées avant que le délai ne soit dépassé.

## 8. Personnalisation des Vues pour les Équipes (Collaboration)

Scénario : Une équipe Réseau veut une vue adaptée pour gérer ses tickets.

Ce qu'ils font :

- Personnalisation de liste : Ajoute les colonnes IP du serveur, VLAN, et Équipement réseau par défaut.
- Masquage des champs : Dans le formulaire, masquent les champs Coût et Contrat (inutiles pour eux).
- Filtre sauvegardé : Crée un filtre Incidents Réseau non résolus et le partage avec l'équipe.

Valeur ajoutée :

- Efficacité : L'équipe voit uniquement les informations pertinentes pour son domaine.
- Standardisation : Tout le monde utilise la même vue, réduisant les erreurs.

## 9. Intégration avec des Outils Externes (Event Management)

Scénario : Un administrateur veut créer des incidents automatiquement depuis un outil de monitoring (ex. : Nagios).

Ce qu'il fait :

- Configuration d'une intégration : Utilise IntegrationHub pour connecter Nagios à ServiceNow.
- Création de règles : Un événement CPU > 90% dans Nagios déclenche un incident dans ServiceNow.
- Formulaire d'incident : Le ticket créé inclut automatiquement les détails (Serveur, Métrique, Seuil).

Valeur ajoutée :

- Réactivité : Les incidents sont créés en temps réel, sans intervention manuelle.
- Centralisation : Toutes les alertes sont gérées dans un seul outil.

## 10. Reporting et Export pour les Audits (Compliance)

Scénario : Un auditeur a besoin d'un rapport des incidents résolus le mois dernier.

Ce qu'il fait :

- Filtre des incidents : Statut = Résolu et Date de résolution = Mois dernier.
- Export vers Excel : Exporte la liste pour analyse hors ligne.
- Tableau de bord : Crée un rapport visuel avec Performance Analytics pour présenter les tendances.

Valeur ajoutée :

- Conformité : Preuves tangibles pour les audits ou les revues de service.
- Analyse : Identification des tendances (ex. : pic d'incidents les vendredis).



## 9. 6) SN - USER ADMIN

Eg admin group > admin role > user A, B, C

- Users: record in sys\_user table, belongs to 1 or + groups, holds 1 or + roles
- Groups: record in sys\_user\_group table, users with common purpose
- Roles: record in sys\_user\_role, assign role to group, contain access control rules  
eg: security admin, catalog admin, ITIL admin
- Access control:
  - record in sys\_\_security\_acl table, assigned to roles, 1000s, to grant access/permissions  
eg: network group > ITIL role grants access while network role deletes access
  - Access operations: read records, write, delete, run scripts, create reports, etc
- Access control execution
- LDAP:
  - protocole pour interroger les services d'annuaires
  - active directory (service annuaire qui stocke les informations utilisateurs, groupes, politiques),

- LDAP utilisé pour vérifier dans AD si utilisateur actif ou pas, ses infos, groupes, les politiques IT, son mot de passe (authentification)
- SSO
- Impersonation: quand un admin veut se logger avec un profil pour vérifier ce qu'il voit

## 10. 4) SN - CUSTOMIZATIONS

### Client side

#### -UI Policies:

Contrôler ce qu'on affiche ou pas.

Met les champs formulaire en show/hide, obligatoire, read-only

#### -UI actions:

Code javascript côté client qui ajoute des boutons qui font une action

Eg: bouton pour valider une demande de congés qui vérifie les jours restants avant de soumettre

#### -Clients script:

Script js client side

Eg: valider des champs, afficher un message

### Server side

#### -Business rules

Règles métier

Eg: assigner un ticket à un groupe spécifique en fonction de sa priorité

#### -Data policies:

Eg rend note obligatoire en cas de clôture d'incident pour rester compliant

#### -Script includes:

Fonctions accessibles depuis un script via GlideAjax

Eg: pour valider si un user a les droits pour valider une demande de changement

#### -UI actions

#### -Update sets

Déployer des scripts par eg entre les instances de dev -> test -> prod

#### -Plugins

Eg: intégration Slack

## 11. 8) SN - SYSTEM ADMINISTRATOR

### -Configuration de base

Personnalisation de l'identité visuelle de l'instance (ex : logo client, couleurs corporate).

Fuseau horaire, langue par défaut, format de date.

### -Propriétés système

glide.ui.list.page\_size: Définir le nombre de lignes par page dans les listes.

mail.smtp.host: Configurer le serveur SMTP pour l'envoi d'emails pour les notifications

glide.email.max\_attachment\_size: Limiter la taille des pièces jointes.

glide.security.password.reset.force: Forcer la réinitialisation des mots de passe.

glide.ui.list.page\_size: Définir le nombre de lignes par page dans les listes.

glide.security.ip.restrict: Restreindre l'accès à certaines adresses IP.

-Tableaux de bord: Tout utilisateur peut créer un dashboard, mais l'admin doit gérer les permissions et la performance.

-Self-Service: Portail personnalisable pour tous les utilisateurs.

-Mobile: applications natives

-Mises à jour (Upgrades)

Maintenance et nouvelles fonctionnalités.

Demander les upgrades via ServiceNow HI ([hi.service-now.com](http://hi.service-now.com)).

Cloner l'instance avant une upgrade pour tester et documenter les customisations impactées.

#### -Diagnostics système

Informations clés (build, mémoire, sessions actives, temps de réponse).

Utiliser ces données pour identifier les goulots d'étranglement.

#### -Résolution de problèmes/Troubleshooting

Activer les logs de debug (glide.log.level) pour les investigations.

#### -Temps de réponse (Response Time)

3 composantes : Serveur : Traitement des données / Client : Rendu de l'interface. / Réseau : Transfert des données.

Optimiser les requêtes et les scripts pour réduire la latence.

#### -Sécurité

Protection donnée

Restrictions IP, chiffrement

Auditer régulièrement les rôles et les permissions.

#### -Notifications

Par email, SMS

Personnalisation des messages.

Tester les notifications avant déploiement.

#### -Workflows

Approvals, conditions, tâches, timers.

#### -Import Sets

CSV, Excel, web services, bases de données.

Chargement → Transformation → Intégration.

Valider les mappings avant import.

#### -Reporting

Prise décision

Rapports standard, personnalisés, planifiés.

#### -Rôle

Associe utilisateurs et groupes à des permissions.

Moindre privilège

#### -ACL

Contrôlent l'accès aux tables, champs, records.

Tester les ACLs avec des scénarios réels.

## 12. 9) SN - BUILD CUSTOM APPS

**-Technique:** Outils ServiceNow à maîtriser

App Engine: Création et gestion de l'application.

Table & Fields: Définition de la structure de données.

Import Sets: Import de données externes.

Transform Maps: Mapping des données vers les tables cibles.

Workflow Editor: Création des processus automatisés.

UI Policies: Contrôle de l'interface utilisateur.

Business Rules: Logique métier automatique.

Update Sets: Déploiement des personnalisations entre environnements.

-Étapes pratiques pour construire l'app

1. Créer les tables et champs

Définir la structure des données (tables, champs, relations).

Utiliser Table & Fields dans App Engine.

2. Configurer les rôles et groupes

Créer des rôles (ex : game\_admin).

Assigner des utilisateurs à des groupes.

3. Importer les données

Préparer un fichier source (ex : CSV).

Créer un Import Set et un Transform Map.

Exécuter l'import.

4. Personnaliser l'interface

Configurer les formulaires et listes (colonnes, ordre, visibilité).

Ajouter des UI Policies et UI Actions.

5. Automatiser avec des workflows et règles

Créer un workflow pour les approbations.

Ajouter des Business Rules pour la logique métier.

6. Tester et déployer

Tester en environnement de développement.

Utiliser Update Sets pour déployer en production.



## 13. 5) SN - DATABASE, TABLE, FIELDS

-2000 tables and fields

-1 database > table > fields - records stored in tables

-Naming conventions

-Data dictionary tables: schéma de base de données, référentiel qui définit toutes les tables, champs et leurs propriétés

-Schema map: représentation visuelle des tables et relations

-Create/delete tables

-Major tables

User [sys\_user], Group [sys\_user\_group], Role [sys\_user\_role], Location [cmn\_location], Company [core\_company],

Incident [incident], Problem [problem], Change [change\_request], Task [task]

Knowledge [kb\_knowledge], Knowledge Category [kb\_category], Knowledge Base [kb\_knowledge\_base],

Service Catalog [sc\_catalog], Catalog Items [sc\_cat\_item],

Configuration Item [cmdb\_ci], Server CI [cmdb\_ci\_server]

-Fields:

-Globally Unique Identifier

-reference field: clé étrangère pour faire des relations entre les tables

-calculated value: ex âge de l'incident

-attributs: obligatoires, lecture seule, valeur défaut, max length

-type: integer, string, date, GlideList (ex: liste techniciens disponibles), choice list

Champ	Type	Attributs	Valeur par défaut	Calculé ?	Description
Numéro	String	Lecture seule, unique	Généré automatiquement	Non	Identifiant unique de l'incident.
Titre	String	Obligatoire, max 100 caractères	Non	Résumé court de l'incident.	
Demandeur	Reference	Obligatoire	Utilisateur connecté	Non	Utilisateur qui a ouvert l'incident.

Statut	Choice List Obligatoire	Nouveau	Non	État actuel de l'incident (Nouveau, En cours, Résolu, Fermé).
Priorité	Integer	3	Non	Niveau de priorité (1 = Critique, 5 = Faible).
Date de création	Date/Time	Lecture seule	Date/heure actuelle	Quand l'incident a été créé.
Âge de l'incident	Integer	Lecture seule	Oui	Nombre de jours depuis la création.
Description	Text Area	Non	Détails complets du problème.	
Pièce jointe	Attachment	Non	Fichiers joints (captures d'écran, logs).	



## 14. 10) EXERCICES

Synthèse des compétences travaillées

Section	Compétences acquises
<b>Section 3</b>	Personnalisation des listes et formulaires.
<b>Section 4</b>	Business Rules, UI Policies, Client Scripts.
<b>Section 5</b>	Création de tables et gestion des données.
<b>Section 6</b>	Gestion des utilisateurs, rôles et ACLs.
<b>Section 7</b>	Relations entre enregistrements et articles de catalogue.
<b>Section 8</b>	Création de workflows et automatisation.
<b>Section 9</b>	Construction d'une application custom complète.

### 1. Préparation et Découverte

Exercices de base

Créer un compte communautaire ServiceNow : community.servicenow.com

Consulter la documentation produit : docs.servicenow.com

Regarder des tutoriels vidéo : Chaîne YouTube ServiceNow

### 2. Prise en main de l'instance

Exercices de base

Obtenir une instance personnelle ou de démonstration : community.servicenow.com

Se connecter à l'instance et explorer les applications Incident et Problem.

### 3. Personnalisation des listes et formulaires

Exercices de base

Configurer la vue liste des Incidents pour inclure les champs : Number, Caller, Category, Priority

Créer un filtre pour les incidents : Ouverts aujourd'hui ET non assignés.

Sauvegarder le filtre sous le nom "Today's unassigned incidents".

Ajouter le champ "Reassignment count" à la vue par défaut du formulaire Incident.

#### 4. Règles métier et politiques UI

Exercices de base

Créer une Business Rule : Définir le groupe d'assignation d'un Problem à "Administrators" si aucun groupe n'est sélectionné à la création.

Créer une UI Policy : Masquer le champ Priority si Impact et Urgency sont tous deux à 3 - Low.

Activer l'action UI "Save" uniquement pour la vue Incident form.

#### Exercices avancés

Créer une Problem Task : Si un Problem est sauvégarde avec une Priority = 1 - Critical, créer automatiquement une Problem Task liée.

Créer un Client Script : Informer l'utilisateur qu'un Problem avec une Priority = 1 - Critical va créer une Problem Task.

#### 5. Création de tables et gestion des données

Exercices de base

Créer une table "Car" avec les champs : make (Choice), model (String), release\_date (Date), cost (Currency)

Ajouter 5 choix pour le champ make (ex : Toyota, Ford, BMW).

Créer 2 enregistrements de voitures.

Configurer le formulaire à votre convenance et ajouter le champ created au formulaire.

#### 6. Gestion des utilisateurs, rôles et contrôles d'accès

Exercices de base

Créer des utilisateurs, groupes, rôles et ACLs pour le cas d'usage suivant :

3 utilisateurs appartiennent au groupe "Incident Team".

Ce groupe a le rôle "incident\_team", qui permet d'ajouter et modifier les champs Caller et Contact Type sur un Incident.

Créer une nouvelle règle ACL :

Donner la permission de modifier le last name de n'importe quel enregistrement utilisateur.

#### 7. Gestion des relations entre enregistrements

Exercices de base

Créer un Incident A.

Créer un Incident B et le lier à l'Incident A.

Créer une Emergency Change Request.

Depuis l'Incident A, créer un Problem.

Lier l'Emergency Change Request au Problem.

Créer un nouvel article de catalogue "iPhone X".

Créer et publier un nouvel article de connaissance.

#### 8. Workflows et automatisation

Exercices de base

Créer un workflow pour l'article de catalogue créé en section 7.

Ajouter la table sc\_req\_item au workflow.

Ajouter une condition pour que ce workflow ne s'exécute que si cet article est sélectionné.

Ajouter une étape d'approbation et une notification au workflow.

Demander l'article et vérifier la notification dans les logs d'emails.

#### 9. Construction d'une application custom (Rotten Potatoes)

Exercices de base

Créer l'application Rotten Potatoes dans votre instance.

Ajouter un champ "Owned" (true/false) à la table Video Game.

Rendre le champ "comments" obligatoire si un avis reçoit 1 ou 2 étoiles.

Ajouter un champ "Image" à la table Video Game.

#### Exercices avancés

Créer un workflow pour la table Video Game :

Exécuter uniquement pour les jeux non encore sortis.

Envoyer une notification à un groupe de relecture 1 semaine avant la date de sortie.

#### 10. Synthèse des compétences travaillées



