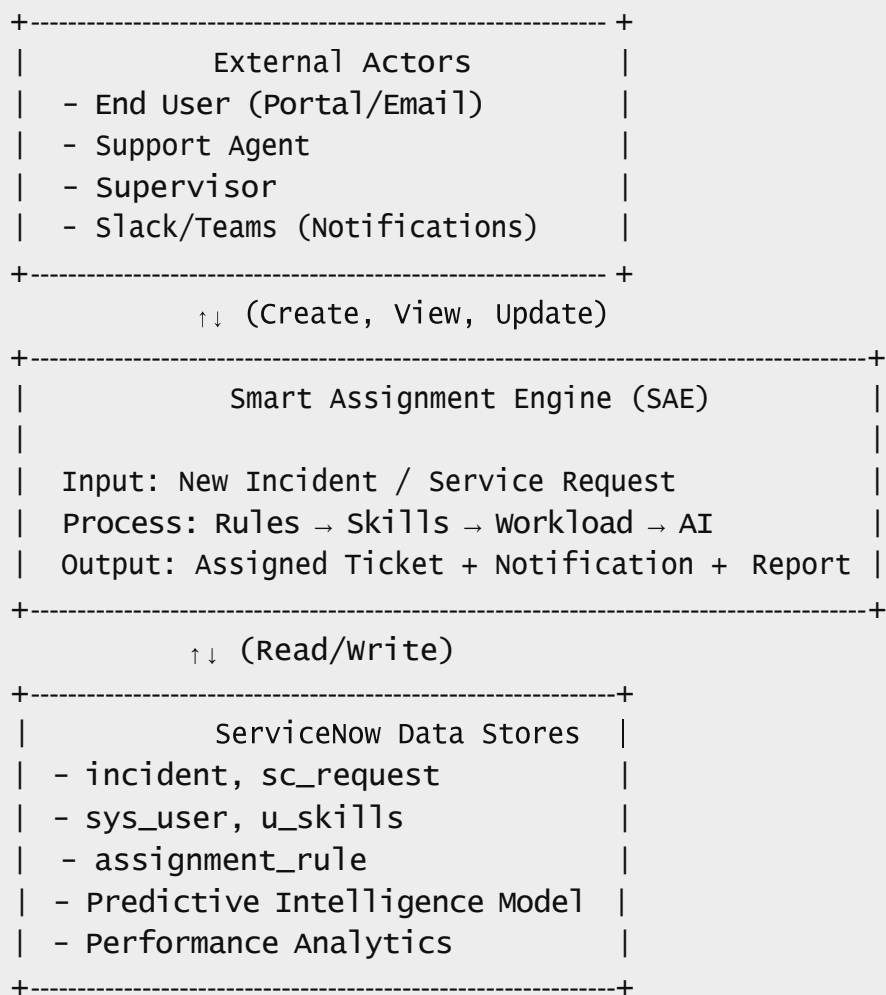


Data Flow Diagrams & User Stories

Date	November 02, 2025
TeamId	NM2025TMID02810
Project Name	Streamlining Ticket Assignment For Efficient Support Operations
Maximum Marks	4 Marks

1. Data Flow Diagram (DFD) – Level 0 (Context Diagram)



2. Data Flow Diagram – Level 1 (Process Breakdown)

```
[New Ticket]
  ↓ (Create)
[1.0 Validate & Categorize]
  ↓ (category, priority, u_required_skill)
[2.0 Apply Assignment Rules] → [Data: assignment_rule]
  ↓ (if match → assigned_to)
[3.0 Skill-Based Matching] → [Data: sys_user.u_skills]
  ↓ (candidate agents)
[4.0 workload Balancer] → [Data: u_current_load]
  ↓ (filter overloaded)
[5.0 Predictive Intelligence] → [ML Model]
  ↓ (suggested assignee)
[6.0 Final Assignment Logic]
  ↓ (set assigned_to, u_assignment_method)
[Assigned Ticket] → [Notify Agent via Slack/Email]
  ↓
[7.0 Update Dashboard] → [PA Indicators]
```

3. Data Dictionary

Data Element	Source	Destination	Description
incident.number	Portal/Email	SAE	Unique ticket ID
category, priority	User input / NLP	Assignment Rules	Routing criteria
u_required_skill	Category mapping	Skill Matcher	e.g., "Network", "SAP"
sys_user.u_skills	HR sync / Self-update	Skill Matcher	List of agent skills
u_current_load	Real-time count	Workload Balancer	Open tickets per agent
predicted_assignee	Predictive Model	Final Logic	ML suggestion
assigned_to	SAE	incident table	Final agent
u_assignment_method	SAE	incident	"Rule", "Skill", "AI"

4. User Stories (Agile Format)

Epic: Smart Ticket Assignment

As a support organization, I want intelligent, automated ticket routing so that agents receive relevant tickets instantly, reducing delays and improving efficiency.

Story 1

As a Support Agent

I want tickets auto-assigned based on my skills

So that I stop receiving irrelevant incidents

Acceptance Criteria:

- ♦ Ticket with `category = Network` → assigned to agent with `u_skills contains Network`
 - ♦ Verified in pilot: 85% skill match rate
 - ♦ Fallback: round-robin if no skill match
-

Story 2

As a Queue Manager

I want real-time visibility into assignment performance

So that I can identify bottlenecks and optimize rules

Acceptance Criteria:

- ♦ PA Dashboard shows: Auto-Assignment %, Avg. Time, Top Skills
 - ♦ Auto-refresh every 5 min
 - ♦ Exportable to CSV
-

Story 3

As a Support Agent

I want to be notified instantly when a ticket is assigned

So that I can start resolution without delay

Acceptance Criteria:

- ♦ Slack/Email notification within 10 sec of assignment
 - ♦ Includes: Ticket #, Short Description, Priority, SLA
 - ♦ Clickable link to ticket
-

Story 4

As a System Admin

I want Predictive Intelligence to suggest the best assignee

So that resolution time improves with historical learning

Acceptance Criteria:

- ♦ Model trained on 6+ months of closed tickets
 - ♦ Accuracy >80% in sandbox
 - ♦ Fallback to rules if confidence <70%
-

Story 5

As a Support Agent

I want workload balancing to prevent overload

So that I'm not assigned new tickets when I have >15 open

Acceptance Criteria:

- ♦ `u_current_load` updated on assign/close
 - ♦ Skip agent if load > threshold
 - ♦ Reassign to next eligible agent
-

Story 6

As a Supervisor

I want a feedback loop after resolution

So that assignment rules can be refined

Acceptance Criteria:

- ♦ Auto-survey: "Was this ticket relevant to your skills?" (Yes/No + comment)
 - ♦ Responses feed into rule refinement sprint
 - ♦ | 70% positive feedback target
-

5. Story Mapping (Prioritized Backlog)

Now (MVP)	Next	Later
Story 1: Skill Matching	Story 4: Predictive AI	Geo-based routing
Story 2: Dashboard	Story 5: Workload Balancer	VIP auto-escalation
Story 3: Notifications	Story 6: Feedback Loop	GenAI categorization

6. Non-Functional Requirements (NFRs)

NFR	Requirement
Performance	Assignment logic <3 sec (95th percentile)
Scalability	Handle 1,000 concurrent ticket creations
Reliability	99.9% uptime; fallback to manual on failure
Security	<code>itil</code> role required to view <code>u_skills</code> ; ACLs on scripts
Auditability	Log <code>u_assignment_method</code> for every ticket

7. Assumptions & Constraints

Assumptions:

- Sufficient and clean historical ticket data is accessible for training the AI models.
- Agent skill profiles will be regularly updated and accurately maintained.
- The Predictive Intelligence module is already licensed and enabled on the instance.

Constraints:

- Solution must be implemented on the ServiceNow Vancouver release or newer.
 - No new custom tables are to be created; only Out-of-Box tables may be extended.
 - Deployment must be completed prior to the peak support period (before December 20).
-