Project Design Phase

Proposed Solution and Architecture

Date	1 November 2025
Team ID	NM2025TMID01446
Project Name	Streamlining Ticket Assignment for Efficient Support Operations
Maximum Marks	4 Marks

Section	**Content**
Proposed Solution	Automate population of assigned_to and assignment_group on Incidents (and other tasks) using native Assignment Rules that evaluate clear conditions (priority, category, location, CI ownership), ensuring consistent routing, reduced triage time, and better accountability without external plugins.
Solution Overview	Drive deterministic, policy-based routing with Assignment Rules; optionally layer Matching Rules and case routing for complex criteria, and trigger Flow Designer automations for notifications and escalations; for skills and capacity-based dispatch, enable Advanced Work Assignment (AWA).
Architecture Goals	Maintain data integrity across incident–user relationships, reduce manual monitoring through rule-driven decisions, and safeguard accuracy with native, ordered rules that run on create/update events.
Key Components	Core elements include the Incident table (assigned_to, assignment_group), Assignment Rules for routing, optional Matching Rules/case routing, Flow Designer for downstream automation, and AWA for load/skills-aware agent selection.

Process flow

A ticket is created or updated with attributes such as category, priority, CI, caller, and location, establishing the context needed for automated routing.
Assignment Rules evaluate in order to set assignment_group and, where appropriate,

assigned_to, producing consistent, auditable outcomes on create/update.
If advanced routing is required, Matching Rules and case routing pre-filter by skills, product, or customer attributes to select the best destination group.
Automation then triggers via Flow Designer for notifications, watchers, and SLA-based escalations; when enabled, AWA allocates work to the best available skilled agent using capacity and presence.
The ticket proceeds through work, potential escalations, and closure, with assignments and transitions traceable to configured rules and flows.

Development phases

Define representative users, groups, schedules, and sample incidents to reflect real routing scenarios across categories, priorities, and locations.Solution-Architecture.pdf
Configure and order Assignment Rules with precise conditions mapped to target groups and, when justified, specific agents, avoiding overlapping or ambiguous criteria.
Introduce Matching Rules or case routing for complex scenarios, and connect Flow Designer to automate notifications, handoffs, and SLA-driven escalations.
Validate behavior end-to-end by testing assigned and unassigned cases, verifying rule precedence, outcomes across edge cases, and error-free handoffs.Solution-Architecture.pdf

Non-functional requirements

Reliability requires deterministic rule evaluation on create and update with clearly documented precedence and safe fallbacks to avoid unassigned records.
Integrity and accountability are reinforced by ensuring assigned_to and assignment_group are consistently populated and valid, preserving incident–user relationships.Solution-Architecture.pdf
Maintainability favors native configuration over custom scripts, enabling simpler operations, upgrades, and ongoing adjustments as teams and services evolve.

Governance and risks

Rule conflicts from overlapping conditions can cause misrouting and assignment churn; mitigate with explicit ordering, narrower conditions, and controlled change processes.
Over-automation may mishandle edge cases; counter with targeted conditions, comprehensive test coverage, and documented manual override guidance.
Operational change can stale routing logic; institute quarterly reviews tied to org structure, services, and support hours to keep rules current.Solution-Architecture.pdf

KPIs and expected outcomes

Time-to-assignment decreases through automation, improving time to first response and accelerating initial triage.

SLA adherence improves as accurate routing and automatic escalations reduce delays and missed thresholds.
Reassignment count and ticket hopping decline due to precise, condition-driven routing and optional skills/capacity-aware allocation.

Scope and assumptions

- Applies primarily to Incident, with a roadmap to extend to other task types (Problem, HR Case, CSM Case) using analogous routing constructs.
- Rules evaluate on create and on update when routing attributes change (e.g., category, CI, location, priority).
- Solution prioritizes native features and avoids external plugins, simplifying adoption and supportability.Solution-Architecture.pdf

Implementation checklist

- Document routing criteria by category, CI/service, location/time zone, and priority, with unambiguous conditions and ownership.
- Build and order Assignment Rules; add Matching Rules and case routing where multi-criteria precision is needed.
- Wire Flow Designer for notifications, watchers, and SLA escalations; pilot AWA where skills/capacity balance is required.
- Test common and edge scenarios; monitor unassigned tickets and reassignments; iterate rules with quarterly governance.Solution-Architecture.pdf

Future enhancements

- Introduce round-robin or load-aware distribution within groups via AWA or controlled Flow Designer logic to balance workloads.
- Add location and schedule awareness for follow-the-sun coverage and proximity-based dispatch when relevant.
- Progress toward ML-assisted recommendations once baseline routing stability and KPI improvements are achieved.