

Solution Architecture

1. Overview

The proposed architecture automates the ticket assignment process within ABC Corporation's ServiceNow environment. It leverages machine learning (TensorFlow) for intelligent ticket classification and Spring Framework for backend integration, ensuring seamless communication between systems and improved ticket routing efficiency.

--

2. Architectural Layers

A. Data Ingestion Layer

Source: ServiceNow incident/ticket records.

Data Collected:

Ticket ID, Category, Subcategory

Description/summary

Priority and urgency

Assigned team or resolver group (historical data)

Resolution time and SLA compliance

Purpose: Provides the input data used to train and test the ML model.

--

B. Machine Learning Layer (TensorFlow)

Model Type: Text classification model using NLP (e.g., LSTM or BERT-based).

Process Flow:

1. Preprocessing: Text cleaning, tokenization, and vectorization of ticket descriptions.

2. Model Training: Trains on historical tickets to learn mapping between ticket content and support teams.

3. Prediction: When a new ticket is created, the model predicts the most relevant team or department.

Output: Recommended team ID or skill group for assignment.

--

C. Application Layer (Spring Framework)

Role: Acts as the middleware between the ML model and ServiceNow instance.

Functions:

API-based communication with TensorFlow model (REST endpoints).

Retrieves new ticket details from ServiceNow in real-time.

Sends prediction requests to the ML model and receives assignment suggestions.

Logs responses and integrates results back into ServiceNow.

Security: OAuth2 or API key-based authentication for secure ServiceNow communication.

--

D. ServiceNow Integration Layer

ServiceNow APIs:

Table API: Fetch and update incident/ticket records.

Scripted REST API: Automate updates when predictions are received.

Automation Logic:

On ticket creation, trigger workflow → call Spring microservice → fetch predicted team → update “Assigned To” field automatically.

Fallback: If confidence level < threshold, route to manual review queue.

E. Monitoring & Feedback Layer

Dashboard: Built within ServiceNow or a Spring UI dashboard to track:

Ticket routing accuracy

SLA compliance improvements

Average resolution time before vs. after automation

Feedback Loop: Agents can manually correct misrouted tickets → corrections are logged → data fed back into model retraining for continuous learning.

3. Workflow Summary

1. Ticket Creation → Logged in ServiceNow.

2. Trigger API Call → Spring microservice fetches ticket details.

3. TensorFlow Model → Predicts appropriate support team.

4. ServiceNow Update → Assigned team auto-updated.

5. Monitoring Dashboard → Tracks assignment accuracy and performance metrics.

6. Feedback Loop → Misassignments retrain model for accuracy improvement.

4. Key Advantages

- ✓ Reduced manual workload for support admins.
- ✓ Faster ticket resolution and better SLA adherence.
- ✓ Smarter workload balancing across teams.

- Continuous improvement through learning feedback.
 - Easy scalability to handle large volumes of tickets.
-