Project Design Phase
Solution Architecture

Date	02 November 2025
Team ID	NM2025TMID04313
Project Name	Laptop Request Catalog Item
Maximum Marks	4 Marks

Architecture Goals and Principles

The primary objective of this architecture is to transform a manual, error-prone process into an efficient, governed, and user-centric digital service delivery model within the ServiceNow platform.

- **Structure & Usability:** Provide an intuitive and guided laptop request system accessible via the Service Catalog.
- **Data Integrity:** Ensure accurate data collection by implementing dynamic form behaviors (validation and visibility).
- **Efficiency:** Reduce manual errors and immediately improve the efficiency of the ITSM fulfillment workflow.
- **Governance & Safety:** Guarantee safe migration and tracking of all configurations using native ServiceNow Update Sets.

Key Architectural Components

The solution is constructed using core, native ServiceNow platform elements to maximize maintainability and scalability:

- **Service Catalog Item:** The front-end user interface, named "**Laptop Request**," categorized under Hardware.
- **Variables/Form Fields:** Structured data capture fields (`laptop_model`, `justification`, etc.) to enforce input quality.
- **Catalog UI Policies:** Business logic for dynamic field presentation. Specifically, to show/hide the `accessories_details` field based on the `additional_accessories` checkbox status.
- **Client UI Action:** A user experience script (named "**Reset Form**") utilizing `g_form.clearForm()` to instantly clear the request form fields.
- **Local Update Set:** The **"Laptop Request"** Update Set, which serves as the governance container for capturing and deploying all configured changes.

Development and Deployment Phases

The solution architecture supports the following ordered development lifecycle:

1. Detailed design of the **Laptop Request** form layout and variable structure.
2. Configuration and testing of the **UI Policies** for dynamic field behavior (conditional visibility/mandatory state).
3. Implementation and validation of the **Client UI Action** (Reset Form).
4. Comprehensive manual testing for accuracy, usability, and adherence to requirements.
5. Completion and export of the **Local Update Set** for migration.
6. Import, preview, and commitment of the Update Set in the target environment.
7. Post-deployment monitoring and preparation for future enhancements (e.g., approval integration).

Solution Architecture Description

The architecture is a streamlined, single-tier deployment focused on the Service Catalog application within the ServiceNow instance.

1. **User Access:** The user navigates to the **Service Catalog** and selects the **Laptop Request** item.
2. **Dynamic Frontend:** The form guides the user through the necessary variables.
    - **UI Policies** execute dynamically on the client-side, adapting the form fields based on real-time user input.
    - The **Client UI Action** provides an on-demand mechanism for the user to quickly correct errors by resetting the form.
3. **Data Submission:** The user submits the request, and a Request (`sc_request`), Requested Item (`sc_req_item`), and Catalog Task (`sc_task`) are generated, initiating the fulfillment workflow.
4. **Governance Layer:** All configuration changes (item, variables, policies, actions) are diligently captured within the **"Laptop Request" Local Update Set** for controlled deployment and auditing.

Solution Architecture Diagram: