1. **ServiceNow Workflow - Mailbox creation Request**

**2. What We Do Once ServiceNow Created the Task (Manual Process)**

**Validation**

* Check that mailbox details meet requirements:
  + Name should not exceed **20 characters**.
  + Name should not contain **special characters**.
  + Ensure there is **no existing mailbox** with the same name.

**Decision Based on User Request**

* Determine the **domain** for mailbox creation (Sainsburys Domain / Sainsburys bank Domain / Sainsburys Hong Kong Domain) based on request.
* Identify if mailbox is **permanent** or **temporary**, and configure accordingly.

**Creation Steps**

* Create the mailbox in **on-prem Exchange** with validated details.
* Add **owners and users** with the required permissions.
* Set mailbox permissions in MS Exchange (Full Access, Send As).

**Post-Creation**

* Send a **confirmation email** to the requester with:
  + Mailbox details
  + Instructions on how to use it
* Update **comments in ServiceNow task** with completion notes.
* Close the request in ServiceNow.

**3. What Automation Does After the Workflow Is Completed**

* Once approval is done, **automation replaces manual mailbox creation**.
* Instead of an IT engineer performing each step, an **API-driven + PowerShell-based automation** handles:
  + Validation of mailbox request details
  + Mailbox provisioning in on-prem Exchange
  + Permission assignment (Full Access, Send As, Send on Behalf)
  + Automatic request closure in ServiceNow

**Audit Logging**

* ServiceNow is updated with a **status comment** for each request:
  + **Failed** → Details on why it failed (e.g., invalid name, duplicate mailbox).
  + **Processed** → Request completed successfully with mailbox details.
* Provides **transparency and traceability** for end-users and audit purposes.

✅ This ensures **consistency, speed, and accuracy** across all mailbox creation requests.

**4. Explaining the Automation (Technical Workflow)**

**1. API Data Fetch**

* An API fetches the entire day’s **mailbox creation request details** from ServiceNow.

**2. Validation & Logging**

* Each request is validated using **advanced PowerShell scripting**:
  + Name must not exceed **20 characters**.
  + Name must not contain **special characters**.
  + No existing mailbox with the same name should exist.
* **Validation Results**:
  + **Failed requests** → Logged in ValidationError.csv (stored securely).
  + **Successful requests** → Logged in ValidEntire.csv for processing.
* All CSV files are retained as an **audit trail** for compliance and future checks.

**3. Processing**

* **ValidationError.csv** → User receives an **error email** with:
  + The reason for failure
  + A link to raise a new request
* *(To reduce rejection rates,* ***help text*** *was* ***recently added*** *in the ServiceNow form with all validation rules and instructions.)*
* **ValidEntire.csv** → Processed automatically via **PowerShell script**:
  + Creates the mailbox in **on-prem** and update in **Exchange**.
  + Assigns required permissions (Full Access, Send As, etc.) in **Exchange.**
  + Sends a confirmation email to the user with mailbox details and usage instructions.

**4. Scheduling & Execution**

* PowerShell script is linked with **Windows Task Scheduler**.
* Runs automatically **daily at 9:00 PM IST / 4:30 PM BST**.
* Processes all requests in one batch.

**5. Automatic Request Closure**

* After successful processing, automation updates ServiceNow via API:
  + Marks request as **Completed**.
  + Adds **completion notes**.
* All mailbox requests are **processed, updated, and closed automatically**.

**5. Advantages of Automation**

* Improved Accuracy (no manual errors in AD updates)
* Transparency (clear ServiceNow audit trail + bulk permissions correctly applied)
* 24x7 Availability (works even on weekends/holidays)
* Better Resource Utilization (engineers focus on complex tasks)
* Reduced SLA Breaches (requests closed consistently on time)

**6. Number of Requests Received in the Last 6 Months**

|  |  |
| --- | --- |
| Months | Task Count |
| Mar-25 | 25 |
| Apr-25 | 30 |
| May-25 | 31 |
| Jun-25 | 18 |
| Jul-25 | 32 |
| Aug-25 | 17 |
| Total | 153 |

Total: 153 requests in 6 months

* Average: 25 requests/month
* With automation, that’s **~ 13 hours saved monthly** (if each manual task takes 30 minutes).