Contents

Project 1: Building of an Automated System for Processing of Requests

Requirements Analysis

1. Inputs and Outputs?
2. Apps to Integrate?
3. Conversion, Transformation, Parsing Process?
4. Decision paths?

Planning the flow and integration

1. Trigger: When a new response is submitted in MS Forms.
2. Setup: Initialize a Variable for 'RequestStatus' and set it to 'Pending'.
3. Data Retrieval: Get the response details. Then, use the 'List rows present in a table' action from Excel Online to retrieve a list of Department Approvers.
4. Data Processing: Use a 'Filter Array' action to find the specific approver from the Excel data that matches the Department selected in the Form. Use a 'Compose' action to extract the approver's email from the array.
5. Parallel Process: Add a Parallel Branch.
   - Branch A: Create a new item in a SharePoint List (or MS Lists) to log the request details.
   - Branch B: Post a message to a specific MS Teams channel notifying the team of a new request.
6. Logic Control: Add a 'Switch' control based on the 'Request Priority' field from the Form.
   - Case 'High': Add a 'Delay' of 1 minute (simulation) before proceeding.
   - Case 'Normal': Proceed immediately.
7. Approval: Start and wait for an Approval using the email found in the Compose step.
8. Decision: Add a 'Condition' to check if the outcome is 'Approve'.
   - If Yes: Send an email via Outlook. The body must use HTML to create a formatted table summarizing the request. Update the SharePoint item to 'Approved'.
   - If No: Update the SharePoint item to 'Rejected' and terminate.

Identifying limitations and possible workarounds

Class Discussion

Project prototype building

Project Discussion	15 mins
Prototype Development	2 hrs
Checking and extension for development duration	45 mins
Troubleshooting and Adjustments	1 hr
[redacted]	

Testing, Checking and Evaluation

Item No.	Skill / Requirement	Where it appears in the Flow	Points	Evaluation
1	Integration with MS Forms	Step 1: Used as the Trigger ("When a new response is submitted").	7.2	
2	Variables	Step 2: Initialize Variable is explicitly requested to track status.	7.2	
3	Integration with Excel	Step 3: List rows present in a table is used to simulate looking up data (e.g., an Approver Matrix).	7.2	
4	Array Controls	Step 4: Filter Array is used to process the raw data from Excel to find a specific row.	7.2	
5	Compose / Data Controls	Step 4: Compose is used to grab a single value (Email) from the filtered array JSON.	7.2	
6	Parallel Branching	Step 5: Explicitly requested to split the workflow into Branch A (Logging) and Branch B (Notification).	7.2	
7	Integration with SharePoint/Lists	Step 5 (Branch A): Used to Create Item for audit logging.	7.2	
8	Integration with Teams	Step 5 (Branch B): Used to Post message to a channel.	7.2	
9	Switch	Step 6: Routes the flow based on "Priority" (High vs Normal).	7.2	
10	Delays	Step 6: Used inside the High Priority case (a common pattern for "cooling off" or reminder periods).	7.2	
11	Approvals	Step 7: The core Start and wait for an approval action.	7.2	
12	Conditions	Step 8: The Condition block handles the Approved vs. Rejected logic.	7.2	
13	HTML Modification in Email	Step 8 (Yes path): The prompt explicitly asks for an HTML table in the Outlook body.	7.2	
14	Integration with Outlook	Step 8: Used to send the final notification.	7.2	

Finalization and Monitoring

Class Discussion

Additional Note:

Here is the HTML code snippet for a clean, table-based layout with inline styles to ensure it renders correctly in Outlook.

Instructions

1. Add a Compose action in your flow (inside the "If Yes" branch, before the "Send an email" action).
2. Paste the code below into the Inputs field.
3. Crucial Step: Delete the text inside the square brackets (e.g., [INSERT DYNAMIC CONTENT HERE]) and select the actual Dynamic Content from your MS Forms trigger (e.g., Responder Email, Submission Time, Description).

The HTML Code

```
<div style="font-family: 'Segoe UI', Arial, sans-serif; max-width: 600px; color: #333;">

 <div style="background-color: #003366; color: #ffffff; padding: 15px; text-align: center;">
  <h2 style="margin: 0;">Operational Request Approved</h2>
  <p style="margin: 5px 0 0 0; font-size: 14px;">Bangko Sentral ng Pilipinas - Internal Automated System</p>
 </div>

 <div style="padding: 20px; border: 1px solid #ddd; border-top: none;">
  <p>Dear Team,</p>
  <p>The following request has been successfully processed and <strong>approved</strong>. Please find the summary of the request details below:</p>

  <table style="width: 100%; border-collapse: collapse; margin-top: 15px;">

   <tr style="background-color: #f9f9f9;">
    <td style="padding: 10px; border: 1px solid #ddd; width: 35%; font-weight: bold;">Requestor Email:</td>
    <td style="padding: 10px; border: 1px solid #ddd;">[INSERT RESPONDER EMAIL HERE]</td>
   </tr>

   <tr style="background-color: #ffffff;">
    <td style="padding: 10px; border: 1px solid #ddd; font-weight: bold;">Submission Date:</td>
    <td style="padding: 10px; border: 1px solid #ddd;">[INSERT SUBMISSION TIME HERE]</td>
   </tr>

   <tr style="background-color: #f9f9f9;">
    <td style="padding: 10px; border: 1px solid #ddd; font-weight: bold;">Department:</td>
    <td style="padding: 10px; border: 1px solid #ddd;">[INSERT DEPARTMENT ANSWER HERE]</td>
   </tr>

   <tr style="background-color: #ffffff;">
    <td style="padding: 10px; border: 1px solid #ddd; font-weight: bold;">Priority Level:</td>
    <td style="padding: 10px; border: 1px solid #ddd; color: #d9534f; font-weight: bold;">[INSERT PRIORITY ANSWER HERE]</td>
   </tr>

   <tr style="background-color: #f9f9f9;">
    <td style="padding: 10px; border: 1px solid #ddd; font-weight: bold; vertical-align: top;">Description:</td>
    <td style="padding: 10px; border: 1px solid #ddd;">[INSERT DESCRIPTION ANSWER HERE]</td>
   </tr>

  </table>

  <p style="margin-top: 20px; font-size: 12px; color: #666;">
   <em>This is an automated message generated by the BSP Power Automate System. Please do not reply directly to this email.</em>
  </p>
 </div>
</div>
```

Project 2: Building of Consolidation of Schedule of COD Mandatory Leaves and Dates of Surprise Assumption Automation

Requirements Analysis

1. Inputs and Outputs?
2. Apps to Integrate?
3. Conversion, Transformation, Parsing Process?
4. Decision paths?

Planning the flow and integration

1. Trigger the flow weekly. Initialize a String variable named "ComplianceStatus" and an Integer variable named "ReviewCycle".
2. In a **Parallel Branch**:
   - Branch A: **Get items** from the SharePoint List named "Mandatory Leave Log" where the status is "Pending".
   - Branch B: **List rows** from the Excel Online table named "SurpriseAssumptionDates".
3. Use a **Compose** action to combine the SharePoint items and Excel rows. Then, use the **Filter Array** action to select only items where the "LeaveDate" is in the next 7 days.
4. Use the **Select** data operation to map the filtered data into specific columns (Employee Name, Date, Department), and then pass this into a **Create HTML Table** action.
5. **Start and wait for an approval** (First to respond) sending the HTML Table to the "Department Head" for review.
6. Add a **Condition** to check if the Approval Outcome is "Approve":
   - **If Yes:**
     - Add a **Switch** action based on the "Department" value.
     - **Case 1 (Treasury):** Post a message to the "Treasury Compliance" **Teams** channel.
     - **Case 2 (Operations):** Post a message to the "Ops Risk" **Teams** channel.
     - **Default:** Send an **Outlook** email to the generic "Audit Team" address. The email body must contain the **HTML Table** modified with a replace expression to add inline CSS borders.
   - **If No:**
     - **Delay** the flow for 1 hour.
     - Send an email to the initiator requesting a revision.
7. At the very end, use a **Compose** action to log the final run timestamp.

Identifying limitations and possible workarounds

Class Discussion

Project prototype building

Project Discussion	15 mins
Prototype Development	2 hrs
Checking and extension for development duration	45 mins
Troubleshooting and Adjustments	1 hr

Testing, Checking and Evaluation

Item No.	Skill	Where it appears in the Flow	Points	Participant's Score
1	**Integration (SP, Excel, Teams, Outlook)**	Trigger (Schedule), Get Items (SP), List Rows (Excel), Post Message (Teams), Send Email (Outlook).	10	
2	**MS Lists & MS Forms**	Covered by the "Get Items" (Lists) and the implicit data source. *Note: Forms is usually a trigger, but for "Consolidation" flows, Scheduled triggers are better. If you strictly need Forms, see the "Alternative" below.*	10	
3	**Approvals**	"Start and wait for an approval" step included.	10	
4	**Parallel Branching**	Specifically requested in Step 2 to fetch SP and Excel data simultaneously (performance optimization).	10	
5	**Variables**	"Initialize Variable" step included at the start.	10	
6	**Array Controls**	"Filter Array" and "Select" data operations are explicitly requested to manipulate the data sets.	10	
7	**HTML Modification**	The prompt asks for "Create HTML Table" and specifically requests a replace expression to inject CSS (a common advanced interview skill).	10	
8	**Conditions & Switches**	Step 6 includes a Condition (Approved/Rejected) and a Switch (Department routing).	10	
9	**Delays**	A 1-hour delay is included in the "If No" branch.	10	
10	**Compose**	Used to merge data and log timestamps.	10	

Finalization and Monitoring

Class Discussion

Additional Note:

In Power Automate, the Create HTML table action produces a very plain table with no borders. To make it professional (like a Banking report), you need to replace the standard HTML tags with "styled" tags using an expression.

The Code Snippet (Expression)

You will use a **Compose** action immediately after your "Create HTML Table" step. Paste this expression into the **Inputs** field of the Compose action:

```
replace(
  replace(
    replace(
      body('Create_HTML_table'),
      '<table>',
      '<table style="border-collapse: collapse; width: 100%; font-family: Arial, sans-serif;">'
    ),
    '<th>',
    '<th style="background-color: #003366; color: white; padding: 10px; border: 1px solid #ddd; text-align: left;">'
  ),
  '<td>',
  '<td style="padding: 8px; border: 1px solid #ddd;">'
)
```

Note: You must change 'Create_HTML_table' in the code above to match the exact name of your action if it has a different name (e.g., Create_HTML_table_2).

How this works:

1. **First Replace:** Finds the <table> tag and adds border-collapse (so you don't get double lines) and sets the font to Arial.
2. **Second Replace:** Finds the header <th> tags and adds a Dark Blue background (#003366 - common for banking), white text, and padding.
3. **Third Replace:** Finds the cell <td> tags and adds light grey borders and padding so the text isn't squashed.

How to implement this in your Flow:

1. Locate the **Create HTML Table** action.
2. Add a **Compose** action immediately after it.
3. Click inside the **Inputs** field of the Compose action.
4. Select the **Expression** tab (usually the fx icon).
5. Paste the code snippet above into the expression bar and click **OK**.
6. **Crucial Last Step:** In your "Send an email" action, do **not** use the *Output* from the "Create HTML Table" action. Instead, use the *Output* from this new **Compose** action.

Project 3: Preparation of Disbursement Vouchers

- Requirements Analysis
- Planning the flow and integration
- Identifying limitations and possible workarounds
- Project prototype building
- Testing, Checking and Evaluation
- Finalization and Monitoring

Other Possible projects:

    - High-Value Transaction Exception Handling
        - A branch manager submits a request to clear a transaction that exceeds the standard daily limit. This requires an audit trail and multi-team notification.
    - New Merchant/Corporate Client Onboarding
        - When a new business client is signed, multiple departments (Legal, IT, Operations) need to act simultaneously to set them up.
    - Loan Application Document Chasing
        - A loan officer marks an application as "Missing Documents." The flow automatically notifies the customer and creates a follow-up task, preventing the application from stalling.
    - Monthly Branch Compliance Certification
        - Every month, branch managers must certify that they have audited their cash drawers and security logs.
- Requirements Analysis
- Planning the flow and integration
- Identifying limitations and possible workarounds
- Project prototype building
- Testing, Checking and Evaluation
- Finalization and Monitoring