

Flow Copilot Master Docs (WIP)

This canvas is the single source of truth we're building live in-chat. I'll keep appending to it as we lock each concept.

1) Assignees (Placeholders)

Concept

Assignees are **placeholders** ("roles") that resolve to a **single email string** at runtime. We do not validate internal/external—only whether someone is an **assignee** and whether they also have **coordinator permissions**.

Assignee resolution options

Default: **Contact TBD** (covers most cases).

- **Contact TBD:** resolved later.
- **Contact:** fixed person for all runs.
- **Workspace initializer:** the person who starts the run.
- **From kickoff form:** value from a kickoff form field.
- **From variable:** value from a flow variable passed at kickoff.
- **Based on rules:** conditional routing (first matching rule wins).
- **Round robin:** rotates among a list of contacts.

Visibility / permissions

- Default assignee view: **assignee experience** (spotlight action)
- Optional per-assignee toggle: **Allow to view other assignees' actions** (role-level visibility)
- Optional per-assignee toggle: **Coordinator** permission (gives coordinator controls for runs they're invited to)
- Note: **Flow-level coordinators** can exist without being assignees.

Junior SE consultative prompts (assignees)

- Who are the roles?
 - Are any roles always the same person?
 - How is the right person determined per run?
 - Does any role need visibility into all actions?
 - Who needs coordinator controls vs just assignee view?
-

2) Kickoff / Trigger

Concept

A flow run can start via: - **Manual**: user clicks **Execute - Kickoff form**: shown before starting (manual or via start link) - **Receive a webhook**: incoming trigger - **Scheduled** - **3rd-party app triggers** (Airtable/Asana/Gmail/etc.) - **Start Link** (post-publish): URL that can start a run; if kickoff form enabled, start link shows the kickoff form

Data availability

- **Kickoff form data** is globally referenceable later in the flow.
- **Flow variables** are for additional data passed during kickoff (especially webhook/system kickoff). Builder defines variable name + type (e.g., text, file). Variables can be asked during execution (manual) or auto-filled (automated).

Defaults

- Default kickoff mode: **Manual (Execute)** unless specified.

Junior SE consultative prompts (kickoff)

- How should this process be started (Execute vs start link vs webhook vs schedule vs integration)?
 - What minimum info must be collected at kickoff to avoid back-and-forth later?
 - If webhook/integration kickoff: what variables must be passed to reference later?
-

3) Action model (cross-cutting)

Common action fields

All actions have: - **title** (required) - **description** (optional; supports variable references) - **attachments** (optional) - **due date** (optional) - **assignees** (one or more; **Decision is single-assignee only**) - **visibility** control (e.g., visible to all assignees toggle)

Multi-assignee completion semantics

Actions may support multiple assignees. Completion semantics vary: - **Approval** supports: one / majority / all, and can be sequential or parallel. - **To-Do** supports: one / majority / all when non-sequential. - **Acknowledgement** supports: one / majority / all when non-sequential. - **Form & PDF Form**: multiple assignees can work, but **any one submission completes**. - **Questionnaire**: **any one submission completes** (optional review can gate). - **Decision**: exactly one assignee.

Execution order vs parallel

Two different concepts: 1) **Flow execution order**: by default actions run in **sequential flow order**. 2) **Skip sequential order (per-step option)**: allows a step to run **in parallel with the previous step**. (Additional branching controls like parallel branch/conditional branch are separate and will be defined later.)

4) Actions inventory (v1)

Approval, Acknowledgement, To-Do, File Request, E-Sign, Form, PDF Form, Decision, Web App, Questionnaire, Custom Action, 3rd-party actions.

5) Action spec: Form (LOCKED)

Supported field types

Basic: single-line text, multi-line text, single-selection, multi-selection, dropdown list, dynamic dropdown, heading, image, file upload, paragraph

Predefined: name, address, email address, phone number, date, number, currency, signature

Completion

- No per-action assignee sequential ordering
- Multiple assignees allowed
- **Any one submission completes the form for all**

Outputs

- Produces structured outputs referenceable later in the flow.

Defaults / guidance

- Assume business already has form OR use placeholder fields unless explicitly specified.
-

6) Action spec: File Request (LOCKED)

Concept

A File Request collects **one or more files** for a single purpose (defined by title + description). Multiple files are allowed and common (e.g., "Proof of ID" → upload passport + driver's license).

Completion

- Uploaders: one or more assignees.
- Sequential order (optional): uploaders can be sequential or parallel.
- **Without review:** completes when required upload is done.
- **With review:** completes only after reviewer completes review.

Review

- Reviewer can be **any assignee**.
- In practice reviewer may also have coordinator permissions, but not required/enforced by the action itself.

Outputs

- Outputs **files only**.
-

7) Action spec: E-Sign (LOCKED)

Concept

E-Sign sends a document for signature.

Signers / order

- Supports **multiple signers**.
- Signing can be configured as **sequential** or **parallel**.
- Completion requires **all required signers**.

Document mutability

- Document is **not changeable** after signing starts.

Outputs

- Outputs **signed document**.
-

8) Action spec: Acknowledgement (LOCKED)

Concept

Acknowledgement is used when an assignee must explicitly acknowledge having read or accepted something (commonly terms, notices, or policies).

Attachments

- Attachments are optional and commonly used.

Require attachment review

- When enabled, the assignee **must open the attachment(s)** before completing the acknowledgement.
- This is a UX guardrail only (no reviewer, no approval).

Completion

- Assignees: one or more.
- Sequential order (optional): if enabled, assignees acknowledge in order.
- If sequential order is disabled, required completion can be configured:
 - **one assignee**
 - **majority of assignees**
 - **all assignees**

Outputs

- Acknowledgement event (no transformed artifact).
-

9) Action spec: Questionnaire (LOCKED)

Concept

Questionnaire is a **lightweight, single-question** input action used for quick questions where a full Form would be overkill.

Question types

- Single selection
- Multiple selection

Completion

- Multiple assignees allowed.
- Non-sequential.
- **Any one submission completes** the questionnaire.

Optional review

- Advanced option to require questionnaire review prior to proceeding.
- If enabled, review completion gates step completion.

Outputs

- Selected option(s) from the question.
-

Prompt Blocks (WIP)

Prompt Block: Form

- Form collects structured data.
- Multiple assignees allowed.
- Any one submission completes the form.
- Produces outputs referenceable by later steps.
- Unless explicitly specified: use placeholder fields; do not add review/reject logic.

Prompt Block: File Request

- File Request collects one or more files for a single purpose.
- Multiple uploaders may be assigned.
- If file review is enabled: upload does NOT complete the step; review completion gates completion.
- Output is files only.

Prompt Block: E-Sign

- E-Sign supports multiple signers.
- Signing order can be sequential or parallel.
- Completion requires all required signers.
- Document cannot change after signing starts.
- Output is signed document.

Prompt Block: Acknowledgement

- Acknowledgement captures explicit acknowledgment.
- Attachments may be required to be opened before completion.
- No reviewer or approval logic.

Prompt Block: Questionnaire

- Questionnaire is a single-question lightweight input.
- Question types: single selection or multi-selection.
- Any one assignee submission completes.
- Optional review can gate completion.

Prompt Block: Flow execution defaults

- Unless explicitly specified, steps execute **sequentially** in flow order.
 - A step may be configured to **skip sequential order** to run in parallel with the previous step.
-

IR / Patch (placeholders)

We'll add the exact schema fields and patch ops as we lock each action type (File Request next).