

Power Automate × Teams 承認フロー

Power Automate の画面を再現した「画面イメージ」と、入力すべき値をセットで載せています。
実際の UI はテナントや時期で少し変わるため、見た目ではなく「アクション名」と「入力項目名」で追ってください。
JSON はすべてそのまま貼り付けできる形で掲載しています。

使い方：① 概要スライドで全体を理解 → ② 各画面イメージで入力箇所を確認 → ③ 実際の Power Automate 画面に同じ値を入れる

作成：2026年4月17日
株式会社ニューコム

実装全体の流れ

Power Automate 画面イメージ

アクション一覧

トリガー

- ・Teams にメッセージを投稿 (V3)
- ・When a user responds to an adaptive card

条件

- ・Teams にメッセージを更新 (V3)
- ・Create item / Update item

フロー デザイナー

トリガー

Teams にメッセージを投稿 (V3)

When a user responds to an adaptive card

条件

Teams にメッセージを更新 (V3)

このフローで作るもの

最小構成

1. トリガー
2. Teams にメッセージを投稿 (V3)
3. When a user responds to an adaptive card
4. 条件
5. Teams にメッセージを更新 (V3)

最初のゴール

- ・カードAを投稿できる
- ・ Approve / Reject / Return を受け取れる
- ・承認後に同じメッセージを書き換えられる

準備するもの

準備項目

トリガー

Forms / 手動 / SharePoint のいずれか

通知先

Teams のチャット または チャンネル

データ保管

SharePoint List (推奨)

申請ID

RequestID (GUID)

状態管理

Status / Version / ReturnCount

おすすめ構成

- ・ SharePoint List を 2 本
 - 1) 申請マスタ
 - 2) 承認ログ

理由

- ・ 最新状態と履歴を分けられる
- ・ 差し戻し理由と SLA を追やすい

未経験者向けの進め方

- 1) まず手動トリガーで作る
- 2) 動いたら Forms/SharePoint に置き換える

Step 1 : トリガーを作成する

Power Automate 画面イメージ

アクション一覧

トリガー

- Teams にメッセージを投稿 (V3)
- When a user responds to an adaptive card

条件

- Teams にメッセージを更新 (V3)
- Create item / Update item

新しいフロー (クラウドフロー)

トリガー

Teams にメッセージを投稿 (V3)

When a user responds to an adaptive card

条件

Teams にメッセージを更新 (V3)

入力する内容

フロー名

Teams承認_単独承認_差し戻し対応

トリガー例

手動でフローをトリガーする

後で置換可

Microsoft Forms / SharePoint

最初は「手動でフローをトリガーする」で十分です。

理由：フォームや List の項目設計に引っ張られず、承認の本体だけ先に確認できるため。

Step 2 : Teams にカードAを投稿する

Power Automate 画面イメージ

アクション一覧

トリガー

- Teams にメッセージを投稿 (V3)
- When a user responds to an adaptive card

条件

- Teams にメッセージを更新 (V3)
- Create item / Update item

フロー デザイナー

トリガー

Teams にメッセージを投稿 (V3)

When a user responds to an adaptive card

条件

Teams にメッセージを更新 (V3)

追加するアクション

検索キーワード

Teams にメッセージを投稿 (V3)

投稿先

チャット または チャンネル

メッセージの種類

Adaptive Card

保持する値

Message ID

重要

このアクションの出力にある Message ID をあとで使います。

→ 承認後に「同じカードを上書き」するため。

Step 3 : カードA JSON を貼り付ける

Teams にメッセージを投稿 (V3) の設定

カードA JSON (そのまま貼付)

投稿先

チャット / チャンネル

メッセージの種類

Adaptive Card

カード本文

下の JSON をそのまま貼る

メッセージID

出力として保持

```
{
  "$schema": "http://adaptivecards.io/schemas/adaptive-card.json",
  "type": "AdaptiveCard",
  "version": "1.5",
  "body": [
    {
      "type": "TextBlock",
      "text": " 🚩 承認依頼",
      "weight": "Bolder",
      "size": "Large"
    },
    {
      "type": "TextBlock",
      "text": "申請タイトル: システムリリース対応",
      "wrap": true
    },
    {
      "type": "TextBlock",
      "text": "申請内容: 4/20 に本番リリースを実施したいです。",
      "wrap": true,
      "spacing": "None"
    },
    {
      "type": "TextBlock",
      "text": "申請者: 藤原",
      "spacing": "Small",
      "isSubtle": true
    },
    {
      "type": "ActionSet",
      "actions": [
        {
          "type": "Action.ShowCard",
          "title": " 🟢 確認する",
          "card": {
```

```
      "type": "AdaptiveCard",
      "action": "submitApproval"
    }
  ],
  "body": [
    {
      "type": "Input.ChoiceSet",
      "id": "approvalResult",
      "label": "判断",
      "style": "expanded",
      "isRequired": true,
      "choices": [
        { "title": "承認", "value": "Approve" },
        { "title": "否認", "value": "Reject" },
        { "title": "差し戻し", "value": "Return" }
      ]
    },
    {
      "type": "Input.Text",
      "id": "comment",
      "label": "コメント(必須)",
      "isMultiline": true,
      "isRequired": true,
      "placeholder": "判断理由や補足を入力してください"
    },
    {
      "type": "Input.Toggle",
      "id": "urgentFlag",
      "title": "緊急対応として扱う",
      "valueOn": "true",
      "valueOff": "false"
    }
  ],
  "actions": [
    {
      "type": "Action.Submit",
      "title": "送信",
      "data": {
```

Step 4 : カード応答を受け取る

Power Automate 画面イメージ

アクション一覧

トリガー

- ・Teams にメッセージを投稿 (V3)
- ・When a user responds to an adaptive card

条件

- ・Teams にメッセージを更新 (V3)
- ・Create item / Update item

フロー デザイナー

トリガー

Teams にメッセージを投稿 (V3)

When a user responds to an adaptive card

条件

Teams にメッセージを更新 (V3)

追加するアクション

検索キーワード

When a user responds to an adaptive card

受け取れる値

approvalResult / comment / urgentFlag / action

判断の元

Input.* の id 名と一致

JSON の対応

- ・ Input.ChoiceSet id=approvalResult
 - ・ Input.Text id=comment
 - ・ Input.Toggle id=urgentFlag
- Flow 側でも同じ名前参照できます。

Step 5 : 条件分岐を作る

Power Automate 画面イメージ

アクション一覧

トリガー

- Teams にメッセージを投稿 (V3)
- When a user responds to an adaptive card

条件

- Teams にメッセージを更新 (V3)
- Create item / Update item

フロー デザイナー

トリガー

Teams にメッセージを投稿 (V3)

When a user responds to an adaptive card

条件

Teams にメッセージを更新 (V3)

条件の考え方

左辺

approvalResult

演算子

次の値に等しい

右辺

Approve

別分岐

Return / それ以外 = Reject

分岐の読み方

```
if approvalResult == "Approve"  
    → 承認完了へ  
elif approvalResult == "Return"  
    → 差し戻しへ  
else  
    → 否認へ
```

Step 6 : 承認後に同じカードを上書きする

Teams にメッセージを更新 (v3) の設定

チーム/チャット	元の投稿先と同じ
メッセージID	Step 2 で保持した Message ID
メッセージ種類	Adaptive Card
カード本文	カードBJSON を貼る

カードBJSON (承認後)

```
{
  "$schema": "http://adaptivecards.io/schemas/adaptive-card.json",
  "type": "AdaptiveCard",
  "version": "1.5",
  "body": [
    {
      "type": "TextBlock",
      "text": "✅ 承認完了",
      "weight": "Bolder",
      "size": "Large",
      "color": "Good"
    },
    {
      "type": "FactSet",
      "facts": [
        { "title": "件名", "value": "システムリリース対応" },
        { "title": "承認者", "value": "@{responder}" },
        { "title": "コメント", "value": "@{comment}" },
        { "title": "緊急対応", "value": "@{urgentFlag}" },
        { "title": "承認日時", "value": "@{utcNow()}" }
      ]
    }
  ]
}
```

Step 7 : 否認と差し戻しを分けて実装する

Reject (否認)

実装

- Status = Rejected
- 承認ログに Action = Reject
- 必要に応じて申請者へ通知

考え方

- その申請はここで終了
- 再申請は別申請扱いでもよい

Return (差し戻し)

実装

- Status = Returned
- ReturnCount = ReturnCount + 1
- LastReturnReason = comment
- 承認ログに Action = Return
- カードCを申請者へ送る

考え方

- 申請を終わらせない
- 同じ RequestID のまま修正して再開する

Step 8 : 差し戻しカードCを申請者へ送る

差し戻し時の設定

投稿先	申請者とのチャット
メッセージ種類	Adaptive Card
カード本文	カードC JSON を貼る
表示理由	LastReturnReason を差し込む

カードC JSON (再申請用)

```
{
  "$schema": "http://adaptivecards.io/schemas/adaptive-card.json",
  "type": "AdaptiveCard",
  "version": "1.5",
  "body": [
    {
      "type": "TextBlock",
      "text": "📧 差し戻されました",
      "weight": "Bolder",
      "size": "Large",
      "color": "Attention"
    },
    {
      "type": "TextBlock",
      "text": "理由: @{{LastReturnReason}}",
      "wrap": true
    },
    {
      "type": "Input.Text",
      "id": "revisedBody",
      "label": "修正内容",
      "isMultiline": true,
      "isRequired": true
    }
  ],
  "actions": [
    {
      "type": "Action.Submit",
      "title": "再申請",
      "data": {
        "action": "resubmit",
        "requestId": "@{{RequestID}}"
      }
    }
  ]
}
```

Step 9 : 再申請 (resubmit) を受けてフローを再開する

Power Automate 画面イメージ

アクション一覧

トリガー

- ・Teams にメッセージを投稿 (V3)
- ・When a user responds to an adaptive card

条件

- ・Teams にメッセージを更新 (V3)
- ・Create item / Update item

差し戻し後の再申請フロー

トリガー

Teams にメッセージを投稿 (V3)

When a user responds to an adaptive card

条件

Teams にメッセージを更新 (V3)

再申請時に更新する値

条件

action = resubmit

Body

revisedBody

Version

Version + 1

Status

Pending

RequestID

元の値を継続

ここでやること

- ・修正内容を保存
- ・同じ申請として承認を最初から再開
- ・承認ログに Action = Resubmit を残す

Step 10 : SharePoint List を作る (推奨)

申請マスタ List (主な列)

RequestID	GUID
Status	Draft / Pending / Returned / Completed / Rejected
Version	Number
ReturnCount	Number
Title / Body / Applicant	申請情報
LastReturnReason	直近の差し戻し理由
MessageID	Teams のメッセージ ID

承認ログ List (主な列)

RequestID / Version	どの申請・どの版か
Approver / ApprovalLevel	誰が・どの段階か
Action	Approve / Reject / Return / Resubmit / Remind
Reason / Comment	理由・コメント
RequestTime / ResponseTime	依頼時刻・応答時刻
DurationMin / SLAStatus	時間・SLA

Step 11 : 複数承認 (順次+並列のハイブリッド)

Power Automate 画面イメージ

アクション一覧

トリガー

- Teams にメッセージを投稿 (V3)
- When a user responds to an adaptive card

条件

- Teams にメッセージを更新 (V3)
- Create item / Update item

level ごとに承認を進めるフロー

トリガー

Teams にメッセージを投稿 (V3)

When a user responds to an adaptive card

条件

Teams にメッセージを更新 (V3)

考え方

承認者配列

```
[{"user":"kacho1","level":1}, ...]
```

level=1

Apply to each 並列 ON

条件

全員 Approve なら次へ

level=2

部長へ単独でカードA送信

ポイント

JSON は同じカードAを使い回します。
違うのは「誰に送るか」と「Flow の回し方」
だけです。

Step 12 : 期限切れ再送と SLA 記録

期限切れ承認者だけ再送

手順

1. 承認依頼時に requestTime / dueTime / status=Pending を保存
2. 1時間ごとの定期フローを作る
3. Pending の中から utcNow() > dueTime を抽出
4. 該当者だけカード再送 or 警告カードへ更新

例

```
dueTime = addHours(utcNow(), 24)
```

SLA を残す

保存する値

- requestTime
- responseTime
- durationMin
- SLAStatus

Power Automate 例

```
durationMin = div(sub(ticks(responseTime), ticks(requestTime)), 600000000)
```

見方

- 承認者別平均時間
- SLA 超過件数
- 差し戻し率

画面イメージと実際の入力値の対応表

対応表

1. Teams にメッセージを投稿 (V3)
 - メッセージ種類 = Adaptive Card
 - 本文 = カードA JSON
 - 保持 = Message ID
2. When a user responds to an adaptive card
 - 受信 = approvalResult / comment / urgentFlag / action
3. 条件
 - Approve → カードB
 - Return → カードC
 - その他 → Reject
4. Teams にメッセージを更新 (V3)
 - メッセージID = Step 2 の値
 - 本文 = カードB JSON
5. 差し戻し時の再申請
 - action = resubmit
 - revisedBody を保存
 - Version + 1 / Status = Pending

最終チェックリスト

- カードA JSON を Teams にメッセージを投稿 (V3) に貼った
- Message ID を保持した
- When a user responds to an adaptive card で approvalResult / comment / urgentFlag を受け取れた
- Approve / Reject / Return の分岐を作った
- 承認後に Teams にメッセージを更新 (V3) でカードBへ置き換えた
- Return 時に LastReturnReason を保存し、カードC を申請者へ送った
- resubmit 時に revisedBody を保存し、Version を +1 した
- SharePoint List (申請マスタ / 承認ログ) を作った
- 期限切れ再送と SLA は、基本動作確認後に追加する

おすすめ順：単独承認 → 差し戻し → SharePoint 記録 → 複数承認 → 期限切れ再送 → SLA の順で足すと失敗しにくいです。

Teams の想定表示画面を追加

この追加パートでは、各 JSON が Teams 上でどう見えるかを「想定表示画面」として示します。実装者は Power Automate の設定だけでなく、承認者・申請者にどう見えるかも確認できます。対象は 3 画面：①承認入力カード ②承認後の完了カード ③差し戻し再申請カードです。

使いどころ

- ・ Step 3 のカードA JSON を貼ったあと → Teams では次の「承認入力画面」のように見える
- ・ Step 6 の更新後 → Teams では「承認完了画面」のように見える
- ・ Step 8 の差し戻し後 → Teams では「再申請画面」のように見える

Teams 想定画面①：承認入力カード

Microsoft Teams

アクティビティチャット

承認フロー Bot | 承認依頼

承認フロー Bot 09:32

承認依頼

申請タイトル：システムリリース対応
申請内容：4/20 に本番リリースを実施したいです。
申請者：藤原

判断
 承認 否認 差し戻し

コメント（必須）
[判断理由や補足を入力してください]

緊急対応として扱う

送信

確認する

この画面の見方

Power Automate 側との対応

- ・ 「 確認する」
→ Action.ShowCard
- ・ 承認 / 否認 / 差し戻し
→ Input.ChoiceSet id=approvalResult
- ・ コメント欄
→ Input.Text id=comment
- ・ 緊急対応チェック
→ Input.Toggle id=urgentFlag
- ・ 送信
→ Action.Submit
data.action=submitApproval

Step 4 では approvalResult / comment / urgentFlag を受け取ります。

Teams 想定画面②：承認完了カード

Microsoft Teams

アクティビティチャット

承認フロー Bot 一般
申請通知
リリース相談

承認フロー Bot | 承認結果

承認フロー Bot 09:40

承認完了

件名：システムリリース対応
承認者：田中 部長
コメント：問題ありません。
緊急対応：false
承認日時：2026/04/14 09:40

この画面の見方

Power Automate 側との対応

- ・ この画面は「新規投稿」ではなく既存メッセージの上書きです。
- ・ 使う値
@{responder}
@{comment}
@{urgentFlag}
@{utcNow()}
- ・ 目的
入力欄を消し、結果だけを表示する
- ・ 再送信防止
Input.* と Action.Submit を入れない

Step 2 で保持した Message ID を使って同じ Teams メッセージを更新します。

Teams 想定画面③：差し戻し再申請カード


Microsoft Teams

アクティビティチャット

承認フロー Bot 一般
申請通知
リリース相談

申請者とのチャット | 差し戻し

承認フロー Bot 09:45

 差し戻されました

理由：影響範囲の記載が不足しています。

修正内容
[修正内容を入力してください]

[再申請](#)

この画面の見方

Power Automate 側との対応

- ・ 理由表示
`@{LastReturnReason}`
- ・ 修正内容欄
`Input.Text id=revisedBody`
- ・ 再申請
`Action.Submit data.action=resubmit + data.requestId=@{RequestID}`
- ・ Step 9 でやること
revisedBody を保存し、
Version を +1、Status を Pending に戻す

差し戻しは申請を終わらせず、「同じ RequestID のまま再開」するのがポイントです。

Power Automate 手順と Teams 表示の対応表（追加版）

対応表

Step 3 : カードA JSON を投稿

→ Teams 想定画面①「承認入力カード」

Step 4 : カード応答を受信

→ Teams で承認者が送信した approvalResult / comment / urgentFlag を取得

Step 6 : カードB JSON でメッセージ更新

→ Teams 想定画面②「承認完了カード」

Step 8 : カードC JSON を申請者へ送信

→ Teams 想定画面③「差し戻し再申請カード」

Step 9 : resubmit を受信

→ Teams で申請者が修正して再申請した内容を保存し、フローを再開