プロンプト例・改善事例付き

AIに正確な業務フローを書かせる プロンプト設計術

入力項目・役割・条件分岐・例外処理の伝え方



Start Challenge Consulting co., ltd.
挑む力が、世界を照らす

Start Challenge
Consulting co., ltd.

挑む力が、世界を照らす

Confidential

はじめに ― AIによる業務フロー作成の前提

AIが誤った業務フローを生成する主因は、業務知識の不足ではなく**入力情報の不足**である。
フローの精度は、AIへの情報の渡し方を設計した時点でほぼ決まる。

期待｜作成工数の削減

ヒアリング内容の構造化やフロー図のたたき台作成を、AIによって短時間で行える。

現実｜実態との乖離

曖昧な指示では、実際には存在しない工程や承認ルールを含むフローが生成される。

原因｜情報の不足

AIの知識不足ではなく、渡す情報（範囲・役割・条件・例外）の不足が誤りを生む。

正確な業務フローを得るには、AIへ渡す情報そのものを設計する必要がある

本資料で学べること

- 正確なフロー生成に必要な7つの情報
- プロンプトの基本構造と記述順序
- 条件分岐・例外処理の伝え方
- 出力形式の指定とレビュー観点

注記：解釈＝AIの誤りは入力情報の不足に起因する場合が多い／提案＝情報の渡し方を設計し、不明点はAIに質問させる。

AIで業務フローを作成するメリット

AIの効果は「図を描く手間の削減」ではなく、**構造化・抜け漏れ発見・認識合わせ**の高速化にある。
作成後の合意形成と改善検討までを一連の作業として短縮できる。

01 ヒアリングの構造化

議事メモや口頭説明を、工程・担当者・条件の単位へ短時間で整理できる。

02 抜け漏れの発見

現状業務の未定義箇所（判断基準・例外時の対応など）を洗い出せる。

03 たたき台の即時作成

白紙から始めず、修正前提の初版を用意して議論を開始できる。

04 関係者の認識合わせ

文章ではなく共通の図・表を前提に、部門間の認識差を確認できる。

05 改善前後の比較

As-Is と To-Be を同じ粒度で並べ、削減対象の工程を特定できる。

06 記法への変換

Mermaid などのテキスト記法へ変換し、版管理と再編集を容易にできる。

注記：事実＝AIはテキストの構造化と記法変換が可能／解釈＝効果は作成工数より合意形成の速さに現れる／提案＝たたき台作成を前提に運用する。

AIが作成した業務フローで起きやすい問題

頻出する不具合は7類型に整理でき、いずれも**AIによる推測補完**が最終的な原因となる。
類型ごとに、プロンプト側で補うべき情報が特定できる。

#	発生する問題	業務への影響	プロンプトで補う情報
1	実際には存在しない工程が追加される	現場実態と不一致になる	作業手順の列挙＋推測禁止
2	誰が担当する作業なのか分からない	責任の所在が不明になる	役割一覧と権限・責任範囲
3	承認条件や判断基準が曖昧になる	運用時に判断がぶれる	判断者・数値基準・分岐先
4	正常系だけで例外処理が書かれない	障害・不備時に手順がない	想定例外と対応者・通知先
5	システム処理と人の作業が混在する	自動化対象を判別できない	レーン分離の指示・利用システム
6	開始条件と終了条件が定義されない	フローの範囲が毎回変わる	トリガーと完了状態の定義
7	**AIが一般的な業務を推測して補完する**	**1〜6の根本原因となる**	**制約条件＋不明点の質問指示**

7類型はいずれも「渡していない情報」に対応する ― 次章で必要情報を定義する

注記：解釈＝問題類型と不足情報は1対1で対応する／提案＝類型を点検表として用い、プロンプト作成時に該当情報の有無を確認する。

正確なフロー生成に必要な7つの情報

7項目のいずれかが欠けると、その箇所はAIの推測で補われる。
不明な項目は空欄にせず、**「不明」と明示してAIに質問させる**ことで推測を防ぐ。

1. 業務の目的

何のためのフローか。現状把握か改善検討かで粒度が変わる。

2. 開始条件・終了条件

どの事象で始まり、どの状態で完了するか。範囲を固定する。

3. 入力情報

項目名・必須任意・入力者・入力元。工程の起点を規定する。

4. 登場する役割

作業者・確認者・承認者・システム。責任範囲まで含める。

5. 作業手順

1工程ごとに「誰が・何を使って・何を行い・どの状態で完了か」を示す。

6. 条件分岐

判断者・判断基準・分岐後の処理をセットで渡す。基準は数値で示す。

7. 例外処理

不備・障害・期限超過の検知方法と対応者。差し戻しも含める。

不明な項目は「不明」と明記し、推測ではなく確認質問を求める

注記：解釈＝欠落項目はAIの推測対象になる／提案＝7項目をチェックリストとして用い、不足は「不明」と明示する。

プロンプト設計の基本構造

プロンプトは8ブロックの順序で構成する。**前段で条件を固定し、後段で出力と制約を規定する**ことで、
同じ業務に対して再現性のあるフローが得られる。

順序	ブロック	記載する内容	省略した場合に生じる事象
1	**役割**	AIに担当させる立場（業務分析・業務フロー設計の専門家など）	記述の粒度が一定しない
2	**目的**	何のためのフローか（現状把握・改善検討・引継ぎ・システム化検討）	不要な詳細が混入する
3	**前提**	対象業務・組織構成・利用システム・対象外範囲	一般的な組織像で補完される
4	**入力情報**	業務内容・関係者・入力項目・既存資料の内容	工程の起点が曖昧になる
5	**作成ルール**	表記方法・詳細度・人とシステムの区別・工程番号の付与	粒度が工程ごとにばらつく
6	**出力形式**	工程一覧表・箇条書き・Mermaid・スイムレーンなど	後工程で再利用できない
7	**制約**	推測禁止・一般慣行での補完禁止・不明点は「要確認」と明示	架空の工程・承認ルールが混入
8	**確認**	作成前に不足情報を質問させる（回答後に本作成へ進ませる）	不足箇所が推測で埋まる

注記：提案＝1～8の順序でテンプレート化し、業務ごとに4（入力情報）のみ差し替える運用とする。

業務の目的・範囲の伝え方

目的と範囲は**「含める範囲」と「含めない範囲」を対で指定**することで初めて固定される。
片方のみの指定では、隣接業務がフローへ流入する。

AIへ伝える項目

項目	記載例
対象となる業務	請求書受領～支払完了
作る目的	現状把握／業務改善の検討材料
含める範囲	経理部の受領・確認・支払処理
含めない範囲	請求書発行側の業務／会計監査
現状／改善後	現状フロー（As-Is）として作成
利用する読者	経理部門の実務担当者

範囲の指定イメージ



プロンプト例

「請求書を受領してから支払いが完了するまでの現状業務を整理してください。請求書発行側の業務と、支払い後の会計監査は対象外とします。」

「対象外」を明記しなければ、隣接業務がフローに流入する

注記：例示は説明用の記載例（仮）／提案＝対象・対象外・As-Is/To-Beの3点を冒頭で固定する。

開始条件・終了条件の伝え方

開始条件は**「事象（～したとき）」**、終了条件は**「状態（～になった状態）」**で記述する。
この2点を固定することで、同一業務に対するフローの範囲が毎回安定する。

開始条件（トリガー）｜～したとき

- 顧客から申請を受け付けたとき
- 営業担当者が見積依頼を登録したとき
- 請求書が経理部へ届いたとき
- システムでアラートが発生したとき

終了条件（完了状態）｜～になった状態

- 承認結果が申請者へ通知された状態
- 支払い処理が完了した状態
- 顧客対応履歴がCRMへ登録された状態

※「処理する」等の動作表現では完了判定ができない

開始条件 ― この区間のみをフロー化する（対象外は記載しない） ― **終了条件**

注記：例示は説明用の記載例（仮）／提案＝終了条件は動作ではなく「状態」で定義し、通知・登録の完了までを含める。

入力項目の伝え方

入力項目は項目名だけでなく、**入力者・入力元・未入力時の処理**まで渡す。
未入力時の扱いを示さない限り、AIは不備検知の工程を生成できない。

整理する入力情報（8観点）

- ① 入力項目名
- ② データ形式
- ③ 必須・任意
- ④ 入力者
- ⑤ 入力元
- ⑥ 入力タイミング
- ⑦ 内容の確認方法
- ⑧ 未入力時の処理

⑧は例外処理・差し戻しフローの生成条件となるため、必ず指定する。

入力項目の例（経費申請）

項目	必須	入力者	入力元
申請者名	必須	申請者	社員マスタ
申請金額	必須	申請者	見積書
利用目的	必須	申請者	手入力
添付資料	条件付き	申請者	PDF・Excel

「条件付き必須」は条件そのものを併記する。
例：申請金額が1万円以上の場合は領収書の添付を必須とする。

注記：表は説明用の記載例（仮）／提案＝入力項目は表形式で渡し、必須区分と未入力時処理を必ず併記する。

役割と責任範囲の伝え方

役割名の列挙だけでは担当工程が特定できない。**作業・確認・承認・連絡の権限と責任**まで渡し、
さらに人の作業とシステム処理を分離して指定する。

役割ごとに定義する項目

- 誰が作業するか（実行）
- 誰が確認するか（チェック）
- 誰が承認するか（決裁権限）
- 誰へ連絡するか（通知先）
- システムが自動処理する範囲
- 代理対応が可能か（不在時）

役割の例と権限・責任

役割	区分	権限・責任範囲
申請者	人（実行）	申請内容の入力・訂正・取消
直属上司	人（承認）	10万円未満の承認・差し戻し
部門責任者	人（承認）	100万円未満の承認・予算判断
経理担当者	人（確認）	証憑確認・勘定科目登録・支払
システム管理者	人（保守）	権限設定・障害時の復旧対応
ワークフローシステム	システム（自動）	必須項目チェック・承認者振り分け・通知送信

役割名ではなく、権限と責任まで伝えることが重要

「上司が承認」ではなく「10万円未満は直属上司が承認、不在時は部門責任者が代理承認」と定義する

注記：表は説明用の記載例（仮）／提案＝人とシステムを区分列で明示し、自動処理範囲を独立して渡す。

作業手順を正しく伝える方法

1工程あたり7項目を定型で渡すことで、工程間の粒度が揃う。
記述形式は**「誰が・何を使って・何を行い・どの状態になったら完了か」**に固定する。

1工程ごとに伝える情報

- ① **作業主体**　誰が行うか（人／システム）
- ② **作業内容**　何を行うか（動詞で記述）
- ③ **使用する情報**　参照する項目・帳票
- ④ **使用するシステム**　画面・機能名まで
- ⑤ **作業結果**　生成される情報・状態変化
- ⑥ **次の工程**　誰へ渡るか
- ⑦ **完了条件**　どの状態で完了とするか

推奨フォーマット

誰が・何を使って・何を行い・どの状態になったら完了か

記載例

「経理担当者が会計システムを使用して請求内容を確認し、勘定科目と支払予定日を登録する。」

要素	記載例の該当箇所
誰が	経理担当者
何を使って	会計システム
何を行い	請求内容の確認
完了条件	勘定科目と支払予定日が登録された状態

注記：例示は説明用の記載例（仮）／提案＝工程は同一文型で列挙し、完了条件を必ず末尾に置く。

条件分岐の伝え方

条件分岐は**「判断する人」「判断基準」「分岐後の処理」**を必ずセットで渡す。
「高額の場合」のような相対表現ではなく、金額・期間などの数値で基準を確定させる。

① 判断する人 ＋ **② 判断基準（数値）** ＋ **③ 分岐後の処理**

判断基準（条件）	判断者	分岐後の処理
申請金額が10万円未満	ワークフローシステム	直属上司へ承認依頼を送付
10万円以上100万円未満	ワークフローシステム	部門責任者へ承認依頼を送付
100万円以上	ワークフローシステム	役員へ承認依頼を送付
予算超過	部門責任者	経理部へ確認依頼を実施
必須項目が不足	ワークフローシステム	**申請者へ差し戻し（理由を記録）**

避けるべき記述
「高額の場合は上位者が承認する」

推奨する記述
「100万円以上の場合は役員が承認する」

注記：金額区分は説明用の記載例（仮定値）／提案＝境界値（以上・未満）を明示し、区分の重複と抜けを排除する。

複数条件・優先順位の伝え方

条件が複数ある場合は、条件の内容だけでなく**評価する順番**を指定する。
順番が未指定だと、AIは条件を並列に扱い、同時該当時の挙動が不定になる。

評価順序の指定例（経費申請）

1. 必須項目の不足を確認する
↓
2. 予算の有無を確認する
↓
3. 申請金額を確認する
↓
4. 取引先が新規か確認する
↓
5. 条件に応じて承認者を決定する

プロンプトに含める内容

AND条件とOR条件
「金額10万円以上 かつ 新規取引先」等を明示

条件を評価する順番
不備チェックを最初、承認者決定を最後に置く

複数条件に該当した場合の優先順位
上位条件が成立した時点で後続を評価しない等

どの条件にも該当しない場合の処理
既定処理（デフォルト経路）を必ず定義する

「該当なし」の経路を定義しない限り、フローに行き先のない分岐が残る

注記：例示は説明用の記載例（仮）／提案＝評価順序は番号付きで渡し、既定処理を明記する。

例外処理・エラー処理の伝え方

例外は**「想定する事象」と「事象ごとの6定義」**を対で渡す。
事象のみを列挙しても、検知方法と対応者が無ければ実行可能なフローにならない。

想定する例外

- 必須項目が入力されていない
- 添付ファイルが破損している
- 承認者が不在
- システムへログインできない
- 外部システム連携に失敗する
- 承認期限を超過する
- 同じ申請が重複登録される
- 処理後に誤りが発覚する

「処理後の誤り発覚」は遡及訂正を伴うため、通常の差し戻しとは別に定義する。

例外ごとに定義する内容

定義項目	記載例（承認者が不在）
検知方法	承認待ちが3営業日を超過
対応者	部門責任者（代理承認者）
対応内容	代理承認を実施し理由を記載
通知先	申請者・本来の承認者
再処理の方法	承認後は通常フローへ復帰
記録の保存先	ワークフロー履歴に代理理由を保存

例外は「事象名」だけでは不足 — 検知方法と対応者まで渡して初めて工程になる

注記：表は説明用の記載例（仮）／提案＝発生頻度の高い例外から6項目を定義し、残りは「要確認」として明示する。

差し戻し・再申請・キャンセルの設計

最も見落とされるのは**「フローを戻す処理」**である。
戻す権限・戻す先・再承認の要否を指定し、正常系と差し戻し系を分けて出力させる。

明確にすべき8観点

- 誰が差し戻せるか
- どの工程まで戻すか
- 修正できる項目
- 再申請時に承認をやり直すか
- 申請者が途中で取消できるか
- 承認後に取消する場合の手続き
- 差し戻し理由を記録するか
- 関係者への通知方法

正常系と差し戻し系を分けて出力

正常系フロー

申請 → 確認 → 承認 → 完了

差し戻し系フロー

- 差し戻し：承認者 → 申請者（理由を記録）
- 再申請：修正後、承認を最初からやり直す
- 取消：承認前は申請者、承認後は経理承認が必要

AIには正常系と差し戻し系を分けて出力させる

同一図に混在させると分岐が読み取れず、レビュー時に見落としが発生する

注記：例示は説明用の記載例（仮）／提案＝差し戻し系は別セクションとして出力を指示し、再承認の要否を明記する。

AIに推測させないための制約条件

情報を渡すだけでは不足を推測で埋める挙動は止まらない。**禁止事項を明文化**し、
不明箇所は「要確認」として出力させることで、確定情報と仮説を分離できる。

プロンプトへ追加する指示

- 入力情報にない工程を追加しない
- 一般的な業務慣行で補完しない
- 不明な情報は「要確認」と記載する
- 判断基準が不明な分岐を作らない
- 役割が不明な場合は仮定せず質問する
- システム機能を推測しない
- 確定情報と仮説を分けて表示する

推奨フレーズ

「情報が不足している場合は推測せず、業務フローを作成する前に確認質問をしてください。」

出力の分離イメージ

区分	出力上の扱い
確定情報	渡した情報に基づく工程として記載
要確認	「要確認」と明示し確認事項に列挙
仮説	「(仮)」を付し根拠を併記させる

制約条件と質問指示は必ずセットで置く ― 禁止だけでは空欄が残る

注記：提案＝制約条件は定型文としてテンプレート化し、全業務のプロンプトに共通で挿入する。

出力形式を指定する方法

最初に表形式で内容を確認し、確定後にフロー図へ変換する手順が有効である。
図から始めると、記載漏れの発見と修正指示に手戻りが生じる。

STEP 1　表形式で内容を確認 → 確定後 → **STEP 2　フロー図形式へ変換**

表形式で指定する列（工程一覧表）

① 工程番号	② 担当者
③ 作業内容	④ 入力
⑤ 出力	⑥ 使用システム
⑦ 判断条件	⑧ 例外処理

表形式は行単位で修正指示が出せるため、内容確認の段階に適する。

フロー図形式の選択肢

形式	適する用途
Mermaid	テキスト管理・版管理・再編集
BPMNを意識した構成	システム化検討・要件定義
スイムレーン	部門横断の役割分担の可視化
箇条書きによるフロー	初期確認・ヒアリング時の読合せ
Excel転記用の一覧	既存管理台帳への取り込み

表で内容を確定させてから図へ変換する ― 図から始めると手戻りが増える

注記：提案＝出力形式は用途に応じて指定し、確認段階は表形式、共有段階は図形式を用いる。

そのまま使える基本プロンプト

P6の8ブロックを1つのテンプレートに統合したもの。**〇〇部分を自社業務に置換して使用する。**
作成ルールと出力形式は共通のまま、入力情報のみを業務ごとに差し替える。

あなたは業務分析と業務フロー設計の
専門家です。以下の情報をもとに、
業務フローを作成してください。

【業務名】
〇〇業務

【目的】
〇〇を実現するため

【対象範囲】
開始：〇〇
終了：〇〇
対象外：〇〇

【登場する役割】
・〇〇
・〇〇
・〇〇システム

【入力情報】
・〇〇
・〇〇

【基本手順】
1.〇〇
2.〇〇
3.〇〇

【条件分岐】
・〇〇の場合：〇〇
・〇〇の場合：〇〇

【例外処理】
・〇〇が発生した場合：〇〇

【作成ルール】
・入力情報にない内容を推測して
追加しない
・人の作業とシステム処理を区別する
・条件分岐には判断者と判断基準を
記載する
・例外処理、差し戻し、再申請を
記載する
・不明点は「要確認」と表示する

【出力形式】
最初に工程一覧表を作成し、その後に
Mermaid形式のフロー図を作成して
ください。

使用時の追記（推奨）「情報が不足している場合は推測せず、作成前に確認質問をしてください。」

注記：テンプレートは本資料P5～P17の内容を統合した記載例／提案＝作成ルールと出力形式は固定し、業務ごとに入力情報のみ差し替える。

実践例：曖昧なプロンプトと改善後の比較

同一業務でも、一文の依頼と条件を明示した依頼では出力の質が大きく異なる。
改善後は**7項目の指定によって、推測箇所が「要確認」として顕在化する**。

改善前｜曖昧なプロンプト

「経費申請の業務フローを作ってください。」

発生する問題

- AIが一般的な承認ルールを追加する
- 金額による分岐が不明
- 経理部の確認範囲が不明
- 差し戻しや申請取消が書かれない
- 利用システムが反映されない

改善後｜条件を明示したプロンプト

「申請登録から支払完了までの現状フローを、下記の役割・金額基準・例外条件に従って作成してください。」

追加した指定内容

- 開始・終了条件を指定
- 申請項目を列挙（必須・任意を併記）
- 申請者・上司・部門長・経理の役割を定義
- 金額別の承認基準を指定
- 証憑不備時の差し戻しを指定
- 承認者不在時の代理承認を指定
- 出力形式と推測禁止ルールを指定

注記：例示は説明用の記載例（仮）／解釈＝改善後は不足情報が「要確認」として出力されるため、レビュー対象が明確になる。

生成された業務フローの確認チェックリスト

生成結果は5観点で点検する。**最終確認は現場担当者によるレビューを必須**とし、
AIが推測で補完した工程が残っていないかを実務側の視点で確認する。

① 範囲	② 役割	③ 条件分岐	④ 例外処理	⑤ 正確性
開始条件と終了条件が明確か	各工程の担当者が明確か	判断者が明確か	不備・障害・期限超過が考慮されているか	AIが推測した工程が含まれていないか
対象外の業務が混ざっていないか	人とシステムの処理が区別されているか	判断基準が数値や状態で定義されているか	差し戻し・再申請・取消が定義されているか	現場担当者による確認を実施したか
		分岐後の行き先がすべて存在するか		

①〜④は作成者が点検し、⑤は現場担当者が点検する ― 二段階のレビューを前提とする

注記：提案＝本チェックリストをレビュー票として運用し、⑤の実施記録を残す。

まとめ・お問い合わせ

AIによる業務フロー作成の精度は、**入力情報の具体性で決まる。**
プロンプト設計を標準化し、最終確認を現場レビューで担保する運用が前提となる。

まとめ

- 精度は入力情報の具体性で決まる
- 目的・範囲・役割・入力項目を最初に整理する
- 条件分岐は判断者・判断基準・分岐先をセットで伝える
- 正常系だけでなく例外・差し戻し・再処理も指定する
- 推測させず、不明点は質問させる
- 表形式で確認してからフロー図へ変換する
- 最終的には現場担当者によるレビューが必要

ご支援できること

- 業務フローの可視化・標準化支援
- 生成AIを活用した業務整理支援
- 業務改善・自動化に関するご相談

お問い合わせ
Start Challenge Consulting co., ltd.
担当：平出
info@startchg.com

Start Challenge
Consulting co., ltd.
挑む力が、世界を照らす

注記：本資料の記載例はいずれも説明用の例示（仮）／提案＝プロンプト設計を標準化し、現場レビューを運用に組み込む。

お問い合わせ

業務フローの可視化・標準化、生成AIを活用した業務整理、
業務改善・自動化のご相談を承っております。

ご相談・お問い合わせ先

会社名	Start Challenge Consulting co., ltd.
担当	平出
メールアドレス	info@startchg.com

Start Challenge
Consulting co., ltd.
挑む力が、世界を照らす

Start Challenge
Consulting co., ltd.
挑む力が、世界を照らす

業務フローの現状整理から、生成AI活用の定着までご支援します

本資料の記載例はいずれも説明用の例示です。個別のご相談内容に応じて、御社の規程に沿った形へ具体化いたします。