

1.業務要件定義の目的

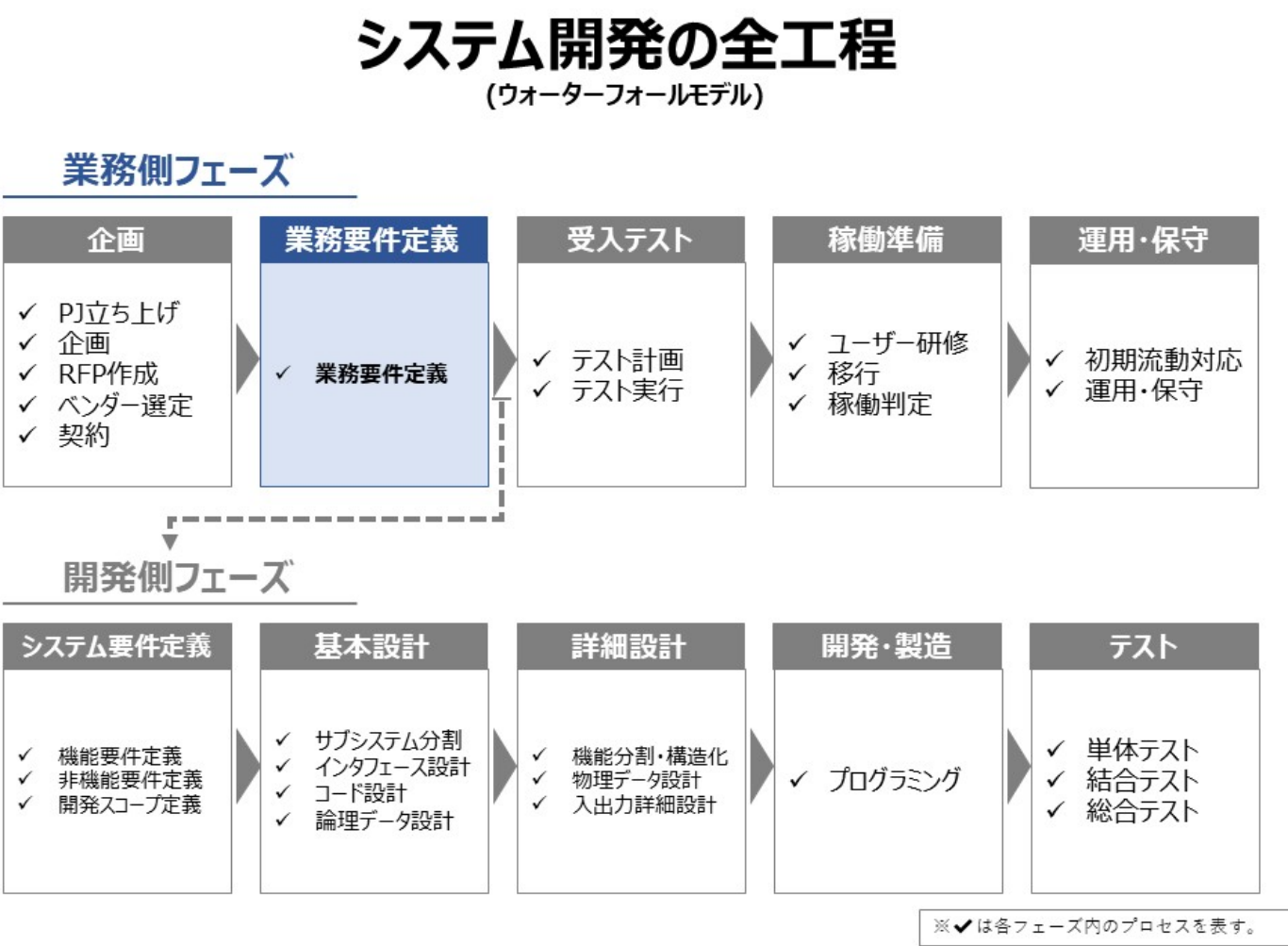
業務要件定義の目的は、下記の通りです。

システム化すべき業務要件を洗い出し、整理することです。

言い換えると、業務要件定義は**現状の業務(AsIs)**を分析し、**あるべき姿(ToBe)**にするための要求を整理することが求められます。

2.業務要件定義の進め方

業務要件定義はシステム開発工程のうち、上流工程にあたるタスクです。(下記の図を参照)



業務要件定義は、システム開発における超重要タスクであり、ここで方向性を誤ってしまうと、後続タスク全てに影響してしまいます。

3.業務要件定義の成果物

3-1.業務/機能要件一覧

業務要件と機能要件が一覧になった文書です。

業務要件と機能要件の定義は以下の通りです。

業務要件：ユーザー側の要件

機能要件：業務要件をシステムの機能に落とし込んだ要件

これらはバラバラに管理するのではなく、業務要件と機能要件を紐づけておくことが重要です。

こうすることで、ユーザー側の要件を抜け漏れ無く機能要件に落とし込んでいるかが一目で把握出来ます。

バラバラに成果物を作成すると、あとで紐づけが大変になるので、始めから同一資料で作るようにしましょう。

3-2.画面定義書

画面定義書とは、「システム画面イメージ」+「機能ごとの作用」を表現した成果物です。

画面上のどの部品がどういう動きをするのかを定義します。

なお、画面定義書の作成方法については、情報処理推進機構のHPが参考になるかと思います。

【参考サイト】

[\(独\)情報処理推進機構 画面レイアウト](https://itconsultant-dictionary.com/rd-process-point/)

システムの項目を定義します。

詳しい項目については、こちらの記事を参考にいただければと思います。

[illegible]

【サンプルあり】 テーブル定義書の書き方を一から解説します

業務ルールがある場合はそちらも成果物として残すことをおすすめします。

一覧にして残しておきましょう。

制約条件は下記**3**つの観点で確認します。

システムや開発における性能条件を押さえておきましょう。

プロジェクトには予算がありますので、青天井にならないよう開発コストの条件も意識する必要があります。

インフラ、とくにネットワーク関連の成約は必ず押さえておきましょう。

3-6.議事録

要件定義書に反映されていない前提条件や背景などを確認するため、議事録はなるべく入手して、読んでおきましょう。

4.要件定義で押さえるべき4つのポイント



4-1：業務フローを叩き込む

要件定義を行うにあたって、まず用意すべき成果物は**AsIs**の業務フロー図です。なぜなら、業務改革を行うためには、現状の業務フローの把握を真っ先にやるべきだからです。

現状の業務を知らない人に業務改革は出来ません。しかし、コンサルタントとしてクライアントの業務改革に入る場合、クライアントより業務内容を把握していない状況からスタートします。

したがって、コンサルタントがまずやるべきは、関係者、タスクの中身、関連システムなどを全て含めた業務フローを頭に叩き込むことです。

業務フローに関しては、プロジェクト関係者の誰よりもコンサルタントが一番詳しくなっているべきなのです。**AsIs/ToBe**業務フローは資料を見なくても頭に浮かべられる状態まで持っていきましょう。

なお、業務フロー作成にあたっては、下記の記事を参考にいただければと思います。



4-2：プロジェクトが発足した目的を常に意識する

プロジェクトの背景と目的を明確にし、そのために何をすべきか、しないべきかを判断するようにしましょう。

また、メンバー間でブレが出ないよう定例会などで意識共有することも重要です。

4-3：要件定義の早い段階で開発側メンバーを入れる

要件定義は必要十分なメンバーで実施する必要があります。特に忘れてはいけないのは、開発側メンバーを要件定義の段階で入れることです。

一般的に、要件定義は業務側メンバーで行い、ある程度要件がまとまった段階で開発側メンバーを入れることが多いです。（つまり、業務要件→システム要件の順）

このとき、注意すべきは、開発側メンバーを入れるタイミングです。

通常、業務側メンバーと開発側メンバーは部署や会社が違うため、コミュニケーションが密に取れる状態でないことが多いと思います。

そのため、コミュニケーションが取れている業務側メンバーのみで要件定義を行い、開発側メンバーに声掛けするのが、要件定義の終盤になるといったような状態になります。

しかし、これは失敗要因の一つです。

提示した業務要件がシステム対応出来ず、もう一度業務要件定義をやり直すといった時間のロスが発生する可能性があります。それを防ぐために、業務要件定義のある程度早い段階で開発側メンバーにインプットすることで、早めの軌道修正を行うことができ、その分ロスも少なくなると言えます。

4-4：業務側と開発側の意思疎通は出来ないと考える

基本的に業務側と開発側の意思疎通は出来ないという前提のもとに要件定義をするべきです。

なぜなら、業務側と開発側はそれぞれ知識、背景、視点が異なるため、同じことを話しても認識が食い違うからです。

システム開発の成否はこの認識の食い違いをどれだけ抑えられるかにかかっています。

そのため、システム開発に必要な成果物は誰が見ても同じ解釈が出来るような形式、文言にすることが重要です。

5.まとめ

いかがでしたでしょうか。

システム開発の上流工程にあたる要件定義は後々の工程に大きく影響を与えます。

要件定義で失敗しないためには、目的・進め方・ポイントなどを押さえて、認識齟齬を出来る限り減らして、後工程に引き渡すことが重要です。