



IT 導入指南

●食品工場

連載ナンバー 003

(株)アップワード
代表取締役
情報システムコーディネーター
●高橋貞二



●たかはし・ていじ
1956年5月20日埼玉
県生まれ。千葉商
科大学付属高校卒業
後、東京急行電鉄(株)
に入社。(株)エスイー
ラボ、(株)インテンシ
ア・ジャパンを経て、
現在、(株)アップ
ワード代表取締役。

要件 定義 PMが プロジェクトの 成否を握る

前回は、「要件分析を行い、IT化の目的を全社の共通認識にすることが成功への秘訣(その1)である」という話でした。今回は、前回の「プロジェクト」の補足説明と、具体的なIT開発の上流工程と呼ばれている「要件定義」について取り上げます。

自社の目標を明確に

IT開発プロジェクトを進める手法は複数ありますが、今回は企業が協力会社(ソフトハウスなど)にIT開発を依頼して、「局面別開発」手法を採用した場合の説明をします(ほかに「プロトタイプ開発」手法がありますが、あまり使用されていません)。

さて、開発の流れとして、①要件分析、②要件定義、③外部(基本)設計、

IT開発プロジェクト成功の鍵になるのです。ここで手法の補足です。「局面別開発」では作業工程の一区切りを「局面」と呼びます。この手法の特徴は、局面の成果物を決定しないと、その局面を終了して次の段階(局面)に進めないということです。ですからこの手法では、要件定義がきちんとできないと、次の外部設計(基本設計)には入れないのです。

PMはプロジェクトのまとめ役

要件定義の説明の前に、一般的なプロジェクトで大切な役目を持つPM(プロジェクトマネージャー)について触れます。物の本によると、プロジェクトとは「独自の成果物やサービスを創造する有期活動(出典||PMBOK)」とあり、次のような特性が挙げられています。

- 明確な「開始」と「終了」がある
- 立場の異なる人や組織が関係する
- 局面ごとに作業内容・作業範囲が異なる
- チームによる共同作業である

●多くの変更や問題が発生する
こういった特性はIT開発プロジェクトにおいても全く同じです。IT技

術を持ち、設計を行うSE(システムエンジニア)やプログラム開発を行うPG(プログラマー)以外で、プロジェクト全体を管理する要員がPMです。プロジェクトに関連する目的・コスト・資源・期限などすべての責任を持つ非常に大事な役目です。PMは企業と協力会社双方に必要であり、このPMがプロジェクトの成否を握るといっても過言ではありません。

PMがプロジェクト開始時に作成するプロジェクト管理計画書は、次のような内容になります。

- 局面完了基準(成果物の決定)
- リスク管理(危機管理)
- プロジェクト進捗管理
- テスト計画
- システム稼働(カットオーバー)基準

要件定義局面

業務をシステム化する時にユーザーの要求をまとめる作業を「要求定義」、その成果物を「要求定義書」といいます。そしてその要求を実現するためにシステム化の要件をまとめる作業を「要件定義」、その成果物を「要件定義書」といい、次の資料などを基に作成します。

●食品工場 IT導入指南

要件定義



表 弊社標準WBS

No.	フェーズ/作業項目	成果物	備考
1	要件定義フェーズ		
1-1	業務要件定義		
1-1-1	業務要求の分析	要件定義書 (課題定義)	何のためにIT化を実施するのか、クリアすべき課題とその解決策を分析する
1-1-2	業務要件の設定	要件定義書 (業務機能要件)	課題解決のために必要な業務要件(機能・帳票・伝票・画面)を決定
1-1-3	業務品質要件の設定	要件定義書 (機能要件マトリクス)	要求事項と現実機能等のマトリクスを作成し、目的に合致しているかどうかを評価
1-1-4	業務基本方針の定義	要件定義書 (開発方針)	リリース方針、テスト工程の考え方等の開発計画の大きく影響する部分の基本方針を決定
1-1-5	方式基本要件の設定	要件定義書 (方式設計基本要件)	ネットワーク、ハードウェア、ソフトウェア、アプリケーション構造、分散形態等の方針を決定
1-1-6	業務要件の顧客合意		
1-2	システム		
1-2-1	業務フローの作成	要件定義書 (業務フロー)	業務の流れを詳細なレベルまで決定
1-2-2	データモデルの作成	要件定義書 (概念データモデル)	データモデル図、エンティティ定義、データ項目定義を決定
1-2-3	入出力概念設計	要件定義書 (入出力概念設計)	物理的な画面帳票ではなく、業務として必要な画面帳票を設計する(画面遷移、ラフスケッチ、入出力項目説明)
1-3	評価と開発方針		
1-3-1	効果と運用費用と顧客運用体制	要件定義書 (効果・費用見積・顧客運用体制)	業務システムを含めた新ビジネスプロセスの生み出す効果とその運用費用の見積を行う。業務運用を実施するために必要な顧客側を中心とした体制を決定する
1-3-2	開発費用見積	要件定義書 (効果・費用見積)	システム開発に要する費用と規模の見積を行う
1-3-3	開発方針の作成	要件定義書 (開発方針・開発日程・体制)	開発基本方針を詳細化し、顧客を中心とした開発日程、開発体制を決定
1-4	システムテストの計画	システム計画書	システムテストの方針、システムテストケース、概要レベルのスケジュールを定義したシステムテストの設計を立てる(必要な期間やリソースの見極めが可能なレベル)
1-5	方式設計と開発準備		
1-5-1	ハードウェア・ソフトウェアの選定	方式設計書 (プラットフォーム構成)	システムを搭載するハードウェア、ソフトウェアを選定する
1-5-2	性能要件の作成	方式設計書 (システム性能設計)	パフォーマンス測定箇所を決定し、性能測定箇所、達成すべき目標値、測定方法を決定
1-5-3	データ量見積	方式設計書 (データ量見積)	システム要求に基づき、トランザクション量、データ量を見積もる
1-5-4	開発標準の作成	方式設計書 (システム開発標準)	システム開発標準(入出力標準、プログラミング標準、エラー処理標準等)
1-5-5	実行環境の設計	方式設計書 (実行処理方式)	アーキテクチャ(2階層、3階層)、DBアクセス、セッション管理、バッチ処理などの実行処理の方式を具体化し決定
1-5-6	処理分散設計	方式設計書 (処理分散方式)	クライアントやサーバー、あるいは拠点別にどのように処理を分散するのか設計する
1-5-7	データ分散設計	方式設計書 (データ分散設計)	クライアントやサーバー、あるいは拠点別にどのようにデータを分散するのか設計する
1-5-8	運用環境と運用方針の設計	方式設計書 (運用方式)	運用の基本方針(サービスレベル要件、運用対象となる機器と運用のためのツール、運用体制、障害対応やバックアップ等の運用業務の方針など)について定義する
1-5-9	環境設計	方式設計書 (環境設計)	本番環境、システムテスト環境(本番環境との差異を明確にした擬似本番環境)、結合テスト環境、開発環境を設計
1-5-10	移行方針の設計	方式設計書 (移行方針)	移行に関わる方針を決定
1-5-11	システム構造設計	方式設計書 (システム構造)	サブシステムの分割、サブシステム構造図、サブシステム内業務機能定義、システム機能一覧など機能分割を行う
1-6	結合テスト方針の作成	結合テスト計画書 (結合テスト方針)	結合テスト方針(目的と範囲、進め方、リスク、環境要件、評価基準、開始終了基準など)を作成する
1-7	システム稼働基準	システム稼働基準書	システム稼働の判断基準を作成する
1-8	顧客の意思決定者承認の確認		
1-9	開発計画書の見直し	開発計画書	
1-9-1	開発計画書の内容修正		
1-9-2	開発計画書の承認		
1-10	要件定義成果物の検証	要件定義成果物の検証記録	
1-11	要件定義の審査		局面終了基準、工程の終了判定(やるべきことはやったか、課題は残っていないか、妥当性確認は完了したか)
1-11-1	要件定義の審査実施	要件定義審査報告書	

- 中長期事業計画書
- 業務内部資料(業務マニュアル/業務フロー/用語集など)
- 業務ヒアリングシート(各部門ごと)
- 業務課題一覧(前回の要件分析ステ

- 現行システムがある場合にはその各種資料(画面・帳票/データベース設計書/仕様書/操作マニュアルなど)

- 打ち合わせ議事録
- 当然ですが、要件定義書のレベルは事前の要件分析がどこまで行われているかで大きく変わります。最近のプロ

プロジェクトでは、要件定義が当初の計画より期間的に長いかかってしまう(半年〜1年)ケースや、要件定義だけでプロジェクトが終了となるケースなどが多くあり、プロジェクトがうまく進



まない例が目立っています。

この理由は複合的なものですが、あえて挙げると、近年の企業の経営環境の変化による業務の複雑化や、もともと要件分析などに重きをおかない企業の体質も関係していると思います。

要件定義の最終決定をするのは企業の皆さまですから、正確で迅速な決断が必要ですが、この判断がうまくいっていません。もし、要件分析を行わず要件定義から開始したプロジェクトがあり、それが上手く進まない場合は、要件定義の作業をいったん止めてでも要件分析を行うことをお勧めします（前号参照）。

要件定義の三要素

ここで少し専門的になりますが、IT開発を請け負う協力会社側から見た要件定義の着目点を3点挙げます。

- ① 業務フロー（依存ダイアグラム／DFD：Data Flow Diagram^{*}）→ システムテストの計画（シナリオ）
- ② 機能構成（サブシステム関連図／機能一覧）→ 機能一覧から概略でのプログラム本数を算出して工数・スケジュール化
- ③ 論理データモデル（ERD：entity-relationship diagram^{*}）→ データベース設計

この3要素をしつかり押さえるこ

と、またこの3方向から要件定義の妥当性を検証して精度を高めていくことが大切です。

逆に3方向から串刺しにして検証しても要件が見えてこなかったり、ぼんやりしている場合は、ペンディング事項など決めるべきことを決め、要件定義を固定させることに集中して下さい。要件定義があまりない状態の基本設計に進んでも結果的に後戻りすることが出てきたり、システム全体の整合性が崩れたり、スケジュールが遅延する可能性が高くなると思して下さい。

WBS

WBS（Work Breakdown Structure）は、プロジェクト管理で計画を立てる際に用いられる手法の一つで、プロジェクト全体を細かい作業に分割した構成図のことをいいます。「各局面での作業項目と成果物の一覧」がWBSです。これは、プロジェクト管理のツールとしてお勧めできます。一部ですが、サンプルとして当社標準の要件定義のWBSの図をご参照下さい。



今回は、各企業の皆さまが実際のプロジェクトで経験したことについて、意見をいただき、それについてお話しします。