

都市整備局・住宅政策本部業務体験発表会
(令和7年度)
概要書

発表テーマ	工事積算システム更新プロジェクトに学ぶ要件定義と社内調整の重要性
発表の概要	<p>当社では、工事積算業務に「積算システム」を使用していますが、OSのバージョン更新により既存システムが動作しなくなるため、新システムへの切り替えが必要となりました。</p> <p>今回の更新では、既存システムと同等の機能を有する「既成のソフトウェア」を採用した結果、当社の積算基準や業務フローに合わせるためのパッケージカスタマイズに多大なリソースを要することが判明しました。</p> <p>一方、社内にはシステムの導入や更新について経験を有する人材が少なく、ベンダーとの協議・調整は困難を極めました。</p> <p>そうした環境にありながら、システム管理課と技術開発・管理課が連携し、試行錯誤を繰り返しながらベンダーと社内ユーザー部署の調整役となって粘り強く協議を繰り返しリリースに至ることができました。(本年11月リリース予定)</p> <p>今回の経験から、要件定義の精度やユーザー部署との連携、社内におけるIT知識の底上げなど、多くの学びを得ることができました。</p>

工事積算システム更新プロジェクトに学ぶ要件定義と社内調整の重要性

1 はじめに

東京都住宅供給公社（以下「JKK」という。）は、受託事業を含め賃貸住宅約 34 万戸の建物維持管理事業を行っている。この管理事業に係る大規模修繕等の工事規模は、年間約 186 億円、契約件数は約 350 件となっており、その多くは、都や区市から受託している公営住宅等である。JKK は政策連携団体として、公共工事の発注者であるという意識を持ち、公正、公平な積算、工事発注の推進に努めている。

一方、昨今特に顕著となっている技術職員の不足は、事業運営上の最重要課題となっており、デジタル化による業務の効率化やミス防止が不可欠となっている。

これまで積算業務は、平成 8 年に開発したシステムを部分的な改修を重ねながら使用しており、OS やミドルウェア等のメーカーサポート停止によるセキュリティリスクや機能不足による手間の多さなど、抜本的な見直しが喫緊の課題となっていた。

そこで、JKK が発注する全ての設計・調査委託、新築、修繕工事等の単価設定と工事費積算を行うための「工事積算システム」を、メーカーのパッケージ製品にアドオン開発を組み合わせる方式で新規開発を行うこととした。



図1 システム再構築イメージ

この開発を進めるに当たっては、パッケージ製品には実装されていない機能について、どこまでアドオンしていくのか等、積算担当者による使い勝手を良くしたいという希望と、それに

係る開発、運用費用との効果検証を重点的に進めた。

その取組として、本システム開発では、上流工程（要求定義・基本設計）を強化し、技術部門を統括している「技術開発・管理課」と社内システムの開発、管理を行っている「システム管理課」が、それぞれ役割を明確にし、社内全体の連携を深めることに努めた。

本稿は、本システム開発の過程で得ることのできたことや反省点を踏まえ、今後のシステム開発をより合理的に進めていくことを目指し、整理したものである。

2 従来システムの活用状況と新たなシステムへの期待

2-1 多様な発注形態で活用するユーザー

2-1-1 従来システムの特徴

JKK が発注する工事の積算は、国や都の基準を準用しつつ、4 職種（建築・土木・機械・電気）に対応している。その中でも土木を除く 3 職種は、さらに「建設」と「修繕」の 2 つの体系を持ち、特に修繕はその工事の特性を踏まえた JKK 独自の積算基準を設けている。

従来のシステムは平成 8 年に開発し、改修を加えながら約 20 年以上にわたり運用しており、職種や発注形態、また、複数棟を発注工区とした大規模建設工事から足場を用いない部分的な塗装補修などの小規模な修繕工事まで適用している。



図2 4 職種の異なる基準体系と多様な発注形態

2-1-2 従来システムにおける主な課題

- ① 開発から 20 年の過程で生じた積算基準の改正や発注方法の多様化等に対し、その都度システム改修による対応を行ってきた。しかしながら、必ずしも全ての変更等に対応できておらず、内容によってはエクセル等のローカル対応を併用してきたところであるが、時間の経過とともに、システムでカバーできないアナログ対応が増加
- ② 職種間で積算基準や単価設定方法が異なり、それぞれの基準改正に合わせて改修を重ねてきたので、システム機能がより複雑化し、全職種共通部分の改修にも影響することもあり、改修や運用費が高コスト化
- ③ WindowsXP 等の旧 Windows OS をベースに開発し、その後のバージョンアップ等に対応するよう改修を行ってきたが、既存ベンダーを含めた検証の結果、Windows11 への改修対応は技術的に困難であることが判明

2-2 新システムの開発にあたって

新システム開発における最大の課題は、要件定義の複雑さであった。

JKK の経営規模では、フルオーダーメイドによる所謂スクラッチ開発は困難であり、可能な限りパッケージに実装されている標準機能を活用していくことがセオリーとなる。しかしその一方で、各職種が長年積み重ねで作り上げてきたノウハウや仕事のやり方を一朝一夕に変更することは困難であり、特に、建築、機械、電気の3職種の修繕工事は、お客様が生活する中で
の工法や手順をきめ細かく定めているので、建設部門の新築工事とは基準が大きく異なる。

こうした状況の中、できるだけパッケージ機能を優先させたいシステム開発部門と従来のやり方をそのままシステム化していくことを希望するユーザー部門との折り合いを付けながら、システム開発事業者との協議を進める必要があった。

そのため、各職種別にチームを編成し、各チームの代表者間での調整等を行う会議体を設置するとともに、積算体系のBPRを実現するため、技術部門の幹部で構成する検討委員会を設置した。今後の積算のあり方を含めた基準の改革を行うことで、パッケージ標準を基本としながら、不可欠な要素や費用対効果の高い機能はアドオン開発を行うなど、適正な開発費とユーザビリティの向上を目指した。

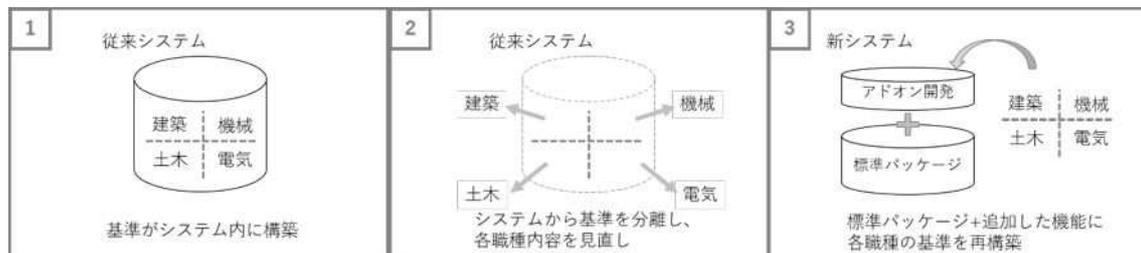


図3 システムと基準の見直しにおける概念図

3 開発に向けての課題と解決策

本プロジェクトにおいて、数多くの課題とそれに向けての取組があったが、その中で特に困難かつ、JKK が実施する他のシステム開発にも共通する課題と解決策を以降に示す。

- ・パッケージシステムのカスタマイズと要件定義の難しさ（3-1）
- ・要求事項のイメージを伝える難しさ（3-2）
- ・開発期間の重要性（3-3）

3-1 パッケージシステムのカスタマイズと要件定義の難しさ（JKK 特有の要件定義）

3-1-1 JKK に求められることの特徴

システム開発では、一般的に「業務の言語化」が重要であると言われ、それが要件定義に直結するとされる。本プロジェクトでは「業務の言語化」の前段で JKK に求められる責任（公共性・公平性・市場競争・合理性・説明責任）を認識することが要件定義の第一歩と考えた。



図4 JKKに求められる責任と工事積算システムの関係

一般企業では、お客様満足度最優先や利益追求、スピード重視等の各々プライオリティがある中で、JKKの工事積算システムに求められるものを意識し、システム開発事業者だけでなく、ユーザー、技術管理部門、IT部門で共有することで、要件定義や開発機能のプライオリティの共通イメージを持つことができる。

3-1-2 異なる積算基準を持つ4職種におけるパッケージ開発の挑戦

JKKでは、前述のとおり4つの職種がそれぞれ異なる積算基準を使用しており、基準の差異が存在する中で、パッケージを利用したシステム開発を進める必要があった。この状況では、パッケージに合わせられる部分とそうでない部分が混在し、パッケージの選定は課題であった。従来システムを単純に更新する場合、4職種と建設と修繕合わせて計7パターンのロジックを維持する必要があり、複雑化のリスクが高かった。

要件定義では、パッケージ適用領域とアドオン領域を明確に線引きするため、Fit&Gap分析を徹底した。その結果、ロジックのパターン化と統一化を進め、積算方式の統一を重要課題とした。また、仕様凍結と変更管理ルールを整備し、追加開発は保守管理コスト、品質、業務への影響を評価したうえで、アドオン領域を必要最小限に抑える必要があった。

3-1-3 システム開発におけるパッケージ活用と要件定義の課題解決

本プロジェクトにおいて、パッケージ選定は発注、契約、監査などを総合的に判断し、最適な製品を選定することが重要な課題であった。候補としては、公共工事に強いパッケージ製品と、建築リフォームに強いパッケージ製品があり、両者の特性を比較検討した結果、官公庁の土木積算基準に合わせたパッケージ製品を採用することを決定した。この判断は、JKKの「建築・土木」基準との親和性が高いことを評価したものである。

しかし、選定後に直面した課題は、建築リフォーム要素の要件定義であった。パッケージを業務に合わせる場合、改修が多くなることは避けられず、既存システムとの整合性も問題となった。従来システムでは「建築・土木」と「機械・電気」に大分され、積算方式も複合単価方式と建材単価方式に分かれていた。両方式には合理性があり一長一短があるものの、従来方式に固執すれば開発費が増大し、工期も延びるリスクが高かった。

そこで、パッケージ仕様を最大限活用するため、積算方式を複合単価方式に統一する方針を採用した。そのため、積算方式は「機械・電気」を「建築・土木」に合わせる方向とし、リリースまでに全ての建材の複合単価を作成する必要が生じた。この対応のため、職種ごとに作業チームを編成し、並行して作業を進めた。さらに、システム開発事業者への指示が職種ごとに自由な意見となるとシステムが複雑化する恐れがあったため、IT部門が調整役を担い、全体の整合性を確保した。

今回のプロジェクトを通じて得られた学びは、パッケージ選定において制度適合性と改修リスクのバランスが極めて重要であるという点である。制度に完全に合わせると改修が増える一方、パッケージ標準を活かすことで開発効率が向上する。また、IT部門の調整機能は不可欠であり、職種間の意見を整理し、システム開発事業者への指示を一本化することで、工期短縮に寄与した。更に、従来業務への固執はコスト増の要因となるため、業務プロセスの見直しを含めたシステム導入が長期的な効率化につながることを再認識した。

3-2 要求事項のイメージを伝える難しさ

3-2-1 イメージ不一致が発生するメカニズム

システム開発では、ユーザーとシステム開発事業者の間で出来上がりのイメージや機能の期待が一致しないことが頻繁に起こる。その背景には、両者の視点の違いがある。ユーザーは業務の流れや使いやすさを重視し、暗黙の前提や期待を持っている。一方、システム開発事業者は機能要件や技術仕様を重視し、仕様書に記載された内容を基に開発を進める。このため、ユーザーが「こういう感じで使いたい」と思っている、それが明確に仕様に反映されなければ、システム開発事業者は対応できない。

さらに、ユーザーは完成品を見て初めて違和感に気づくことが多く、システム開発事業者は要件を満たす機能を優先するため、操作性や業務の自然な流れが後回しになる。この構造的なギャップが、イメージの不一致を生むメカニズムであり、今回のプロジェクトにも頻発した。

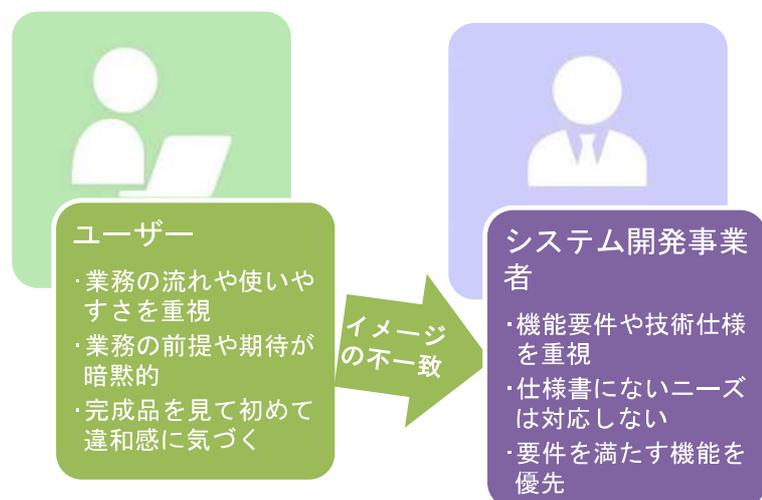


図5 システム開発におけるイメージの不一致

この問題を防ぐためには、プロトタイプやモックアップを用いた早期のイメージ共有、Fit&Gap 分析によるパッケージ適用範囲の明確化、ユーザー参加型レビューの徹底が有効である。そして何より重要なのは、期待や暗黙知を言語化し、仕様として明確にすることである。言語化は、ユーザーの「こうしたい」という感覚を、システム開発事業者が理解し実装できる形に変換する唯一の手段であり、システム開発の成功を左右する鍵となる。

下図は、言語化の難しさを表した有名な説明図である。ユーザーの要求事項は右下の「木にぶらさがったタイヤ」であるが、伝えたのは「木にぶらさがった3段のブランコ」だった。それを聞いたプロジェクトリーダーは右隣と捉え、以下、各々捉え方が違い、全く違ったイメージが出来上がる。本プロジェクトにおいてもイメージの不一致は多く発生し、その都度、何度も認識合わせを行うことに苦勞した。そして不一致に気付かないケースも発生していた。

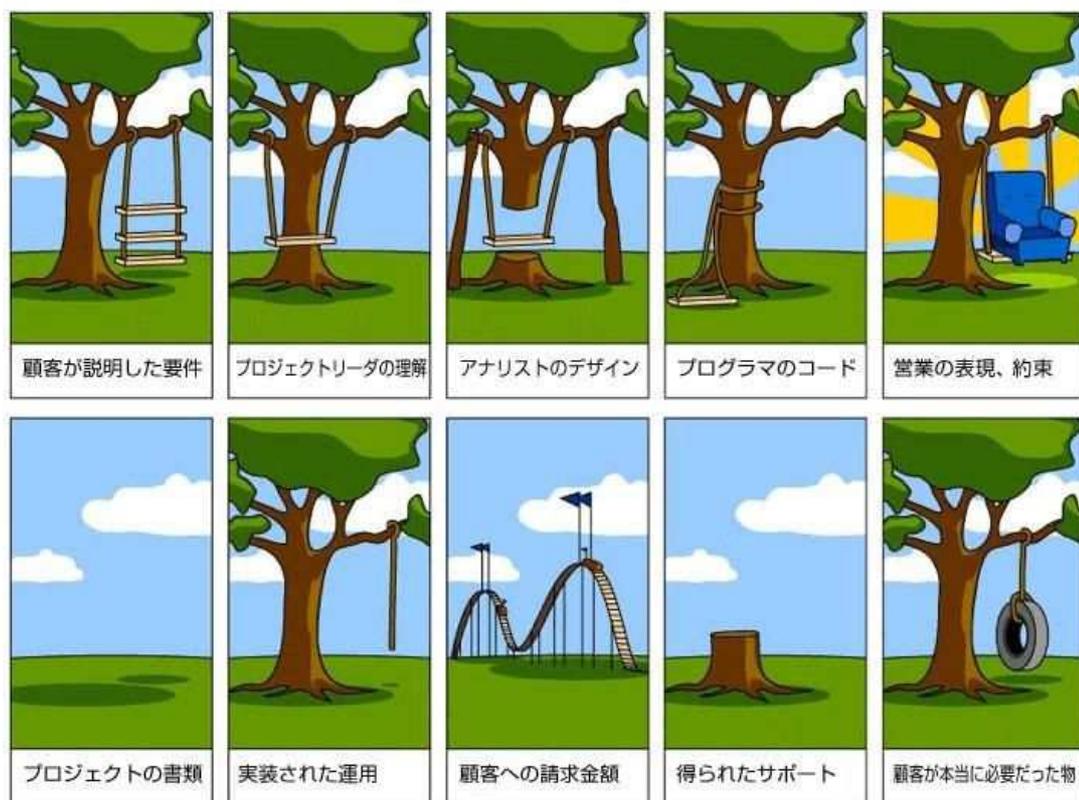


図6 オレゴン大学の実験による顧客要求とその結果の模式図

出展：C. アレクサンダー／宮本雅明「オレゴン大学の実験」(鹿島出版会、1977年)

3-2-2 業務知識が失われている現実

要求事項のイメージを伝える難しさの一つの要素として、業務知識の喪失がある。システム利用の長期化や改修の繰り返しにより、業務知識が失われる現象は多くの組織で見られる。これは単なる人材の異動や退職による知識喪失だけでなく、システムの構造や運用プロセスにも起因する。業務プロセスがシステム化される過程で、知識はシステム内部のコードや設定に埋め込まれ、ブラックボックス化する。業務担当者は、日常的な操作や判断を「システム」で行っていることが多く、ブラックボックス化したプロセスの正確な理解が難しくなる。また、IT

部門も長期間の運用の中で最新のシステムと設計書等が更新されずドキュメントが断片化する。結果として、ユーザー側の業務知識と IT 部門の業務知識が失われ、相互に補完できない状態に陥る。

3-2-3 現行踏襲ではイメージが伝わらない

再構築の際に、「現行踏襲」という要求が発生することが多い。「現行踏襲」を実現する上で注意しなければならないのは、「現行」が何か、ユーザーとシステム開発事業者との間でギャップが発生していないかという点である。

ユーザーは「動作している従来システム」がそのまま踏襲されることを期待しているが、システム開発事業者は開発するには仕様化する必要があるため、設計書などのドキュメントに記載されている要件や仕様、ソースコードに実装されている内容を「現行踏襲」の拠り所の一つとしている。ここにギャップがあるとユーザーとシステム開発事業者のイメージに差が生じる。

【設計書とソースコードのギャップ】

- ・設計書が当初のままで、現行のソースコードが更新されていない

【設計書・ソースコードと実際のシステム動作とのギャップ】

- ・製品として提供されている機能が、設計書に記載されていない
- ・ソースコードでの実装ではなく、運用保守の手動や別ツールなどによる仕組みで、設計書に記載されていない
- ・設計書に記載されているシステムの能力と、実際の能力が異なっている

このようなことにより、どの情報で再構築したいのかを見極めないと、誤った「現行踏襲」になってしまう。

3-2-4 IT 部門の翻訳機能で解決

システム再構築の過程では、イメージの不一致や業務知識の消失、現行踏襲のギャップといった課題は避けられない現象である。こうした課題を解消するためには、IT 部門が業務とシステムの間立つ「通訳」として翻訳機能を果たすことが不可欠である。IT 部門は、要件一覧や変換ルール、テスト観点といった共通言語を整備し、関係者間の認識を統一する基盤を築く必要がある。

さらに、モックアップやプロトタイプを活用して完成イメージを共有することが重要である。本プロジェクトでは、モックアップを十分に活用していなかったが、開発していく中で短周期のプロトタイプを導入することで、認識合わせのスピードを高めることができた。この取り組みにより、ユーザーとシステム開発事業者の間で曖昧だった要件が具体化し、後戻りのリスクを減らすことができた。

また、IT 部門が調整役として機能することで、関係者が自由に意見を述べられる環境が生まれ、潜在的なリスクを早期に顕在化させることが可能になる。こうした役割を果たすことで、IT 部門は単なる技術提供者ではなく、業務とシステムを結びつける橋渡し役として、プロジェクトの成功に大きく貢献すると考える。

3-3 開発期間の重要性（上流工程とスピードが重要）

3-3-1 住宅建設工事請負契約とシステム開発契約の違いから見る上流工程以降の固定化

JKKの本業ともいえる工事請負契約とシステム開発委託の違いから、システム開発の進め方の課題を考える。

工事請負は設計図・仕様書の確定を契約の前提とし、途中変更は契約変更や追加請負となるのに対し、システム開発では要件定義を契約範囲内で確定し、開発フェーズ以降の要件変更が再設計・再開発を招くことになる。

また、工事請負では、設計段階で完成イメージを図面や模型で確認できるため、後戻りを最小化できる。一方、システム開発では図面や模型に代わるものがなく、完成イメージの共有が難しいため、要件定義と基本設計の精度が非常に重要となる。

項目	工事請負契約	システム開発契約
契約タイミング	設計確定後が多い	要件定義前に契約が多い
要件定義（設計）の位置	契約前に完了することが多い	契約中に実施することが多い
変更費用	設計後は高額（撤去・再施工）	開発後は高額（再設計・再開発）
事前確認手段	図面・模型で確認可能	モックアップ・プロトタイプが有効
柔軟性	低い（物理的制約） 撤去が必要となり困難	高い（論理的制約） 要件定義に戻ってしまう

表1 工事請負契約とシステム開発契約の違い

システム開発では、上流の合意形成と変更管理の厳格化が不可欠である。後戻りは、やろうと思えばできてしまい、それにより工数と品質リスクを増やすことになる。

3-3-2 基本設計以降の開発にスピード感を持って（要件定義に戻らない）

システム開発において、基本設計以降のフェーズを迅速に進めることは、次の理由によりプロジェクトの成功に直結する重要な要素である。

第一に、システムの恩恵を早期に享受するためである。システム開発は多額の投資を伴うため、運用開始を早めることで業務効率化や投資回収・コスト削減などの効果を早期に得ることができる。また、建設業界の変化の中で、導入の遅れは協力会社との技術や積算方法の差異につながる。

第二に、開発期間中の制度変更を避けるためである。長期にわたる開発では、制度や業務関連法令の改正が発生する可能性が高まり、追加要件や再設計が必要となり、コスト増や納期遅延の原因となる。このリスクを低減するためには、制度改正の動向を事前に調査し、要件定義に反映させることが重要である。また、基本設計後は迅速に詳細設計・開発・テストを進め、制度変更の影響を最小限に抑えることが必要である。

第三に、人事異動による認識の変化を最小限にするためである。長期プロジェクトでは、担当者や責任者の異動により認識や優先度が変わるリスクがある。これが仕様変更や追加調整を

招き、開発効率を低下させる。この問題を防ぐためには、要件定義書や設計書を最新状態で管理し、関係者全員がアクセスできる環境を整備することが不可欠、さらに、仕様変更や追加要件の承認フローを明確化し、異動による混乱を防ぐ仕組みを構築することが重要である。

基本設計以降のフェーズを迅速に進めることは、ビジネス価値の早期実現、外部環境変化リスクの低減、組織的な認識の一貫性維持という観点で合理的である。そのためには、開発手法の工夫、プロジェクト管理の強化、情報共有の徹底が不可欠であり、これらを組み合わせることで、スピードと品質を両立した開発を実現することができる。

3-3-3 受入テストのジレンマ

システム開発の最終段階である受入テスト（ユーザーテスト）では、本来の目的は要件定義や基本設計の充足確認だが、実際には使い勝手やユーザビリティに関する意見が多く出ることがある。この結果、要件外の改善要求が発生し、対応すれば工期・コストが増加し、対応しなければユーザー満足度が低下するというジレンマが生じる。仕様書上は適合でも、実際の運用に耐えられないような運用のリアリティを満たさないことがあるため、現場で初めて要求との違いが顕在化する。

この背景には、①テスト目的の混在、②変更要求の優先度基準の不明確さ、③改善計画の見える化不足、④フィードバックの質の低さ、⑤コミュニケーション不足といった課題があり、これらが整理されないと、リリース遅延や不信感の増大という負のサイクルに陥ることがある。

解決には、テストの定義を二層化し、受入テストは要件適合確認、UIテストは使い勝手改善と明確に分ける。次に、改善要求は影響度と改修コストで優先度を付け、今回対応・次期対応・保留を仕分け、さらに、改善は次期リリース計画を作成し、進捗が見える化することで、不断の品質改善に努めることが重要である。

この状況は、住宅建設の竣工検査と入居後の住み心地改善に似ている。竣工検査では設計図通りに建っているかを確認するが、入居者からは照明位置や動線などの要望が出る。全てを検査段階で直していたら引渡しが遅れる。そこで、引渡し後のアフターサービスや定期点検で改善していく、修繕計画を公開することで信頼が維持できる。ITシステムでも同じで、検収＝終わりではなく、検収＝改善の始まりという考え方が正のサイクルを生み出すと考える。



図7 受入テスト実施状況

4 まとめ

4-1 要件定義・社内コンサルティング・IT知識の底上げと人材育成

本システム開発プロジェクトを通じて得られた知見、効果は以下となる。

4-1-1 要件定義の重要性

要件定義は、ユーザーのニーズを正確に反映し、システムで何を実現するかを決定する極めて重要な工程である。要件定義でユーザーが必要とする機能を明確に定義することで、初めて業務に合ったシステムが完成する。要件定義が不十分な場合、後続工程において関係者間で認識の齟齬や手戻りが発生し、仕様の変更につながるなど工程の大きなロスとなる。上流工程で決定したことは後続の工程で極力変更せず、後戻りしないことは、プロジェクトの円滑遂行に際する大原則である。

ユーザーへのヒアリングを緻密に行い、現状の業務フローの必要条件や課題と制約を明確化して、機能要件の抜け漏れを防ぐ地道な作業と、機能要件と非機能要件に整理したうえで、社内外の関係者間で合意形成を行い、正式なドキュメントとして承認するプロセスが重要である。

4-1-2 社内コンサルティングの重要性

プロジェクトを円滑に進め、品質の高いシステムを完成するためには、システム開発事業者とユーザー部門とをつなぐための社内コンサルティング機能をいかに発揮できるか肝要となる。ユーザーの希望を取りまとめシステム開発事業者に正確に伝えることや、システム開発事業者の理解にすれ違いないことの確認はシステム開発を担当する部門の責務である。

そのため、今回の開発では、技術を統括する部門とシステム開発担当がタッグを組み、ユーザー部門の仕事の内容を深く理解していくことに努めた。その前提の下、ユーザーが希望している機能は真に効果的なものなのか、他に選択肢はないのか等を検討し提案していく。ユーザー部門の職員も自らの仕事の内容を理解し重く受け止めていると感じることで、協議が円滑に進むことを実感した。

また、システム開発事業者に業務フローやそのやり方を伝えたつもりでも、それが不十分で、理解に齟齬があると完成時に希望した内容と違うというケースがある。住宅設計に例えるなら、出入口は2か所、階段は1か所と伝えたところ、完成した家には玄関は並んで2か所、階段は幅が狭くて想定していた家具が運べないという状況。発注者の意図は、玄関と裏庭に勝手口が必要だった、2階には大型の家具を置きたかったことだとしても、システム開発事業者側は、指示のない事項は受けた側の解釈で設置し、修正するなら撤去費を含む追加費用が必要となる。

これを防止するための対策として、モックアップやプロトタイプの実用は極めて重要であり、開発費や期間が増えることにはなるが、品質を確保するための必要経費であると思料する。

これらに加え、関係部門との役割を明確にすること、ToDoの整理、要件定義書や議事録など情報共有の仕組みを整えることも重要となる。合意形成プロセス（ボトムアップ/トップダウン）のルール設定、会議での論点の明確化と意思決定者の参加、定期的な進捗確認による情報共有など、社内での合意形成のスピードと意思決定の透明性確保を念頭に、プロジェクトを進めることが重要であると認識した。

4-1-3 社内における IT 知識の底上げと育成の重要性

システム開発において、社内の IT 知識を底上げすることは、プロジェクトの成否に直結する重要な要素である。IT リテラシーの向上は、要件定義の精度向上、システム開発事業者との円滑なコミュニケーション、コスト判断力の強化、セキュリティ対策の充実、更には DX 推進の基盤構築に寄与する。

特に、システム開発の成否は要件定義に大きく依存するため、IT 知識の醸成は不可欠である。知識が不足している場合、要件定義において曖昧な表現や現実味の乏しい要求が生じ、後工程で重大な後戻りを招く恐れがある。一方、社内に十分な IT 知識が備われば、こうした問題を未然に防ぎ、プロジェクトを円滑に進めることが可能となる。

このような IT 知識の底上げは、IT 部門がシステム開発プロジェクトを通じて推進することが有効である。JKK は新たに IT 職の職員を採用し、システム開発や運用管理、人材育成に従事させている。本プロジェクトにおいて IT 職の貢献は極めて大きく、今後の DX 推進においても重要な役割を担うものと認識している。

4-2 今回のプロジェクトの効果

4-2-1 本システム開発の主な効果

- ・ 主に窓口センターが発注する伝票発注工事、住宅営繕部が発注する大規模修繕工事や除却工事、建設部門が発注する新築工事等、JKK が行う工事発注の全ての積算に適用することから、人事異動により部門が変わっても、すぐに積算業務に着手可能
- ・ 単価設定や構成方式や積算方式が各職種統一できたことで、今後の制度改正等によるシステム改修や工事項目の職種変更に対応
- ・ パッケージ標準に合わせることで安定稼働に向けた保守性や運用コストの縮減が実現

4-2-2 主なユーザビリティの向上

- ・ プルダウン検索・文字検索対応や画面遷移の簡素化などのユーザーインターフェース改善で作業効率が向上
- ・ 電子決裁システムや契約システムとの連携強化により、証跡確保と取り間違い防止効果が向上
- ・ 大規模修繕の工事種別の組み合わせ（同調化発注）や、総合発注への対応が可能となり、不調対策に向けた発注ロットの変更に対応が実現

これらは主な効果として記載したものであるが、今後、各部門にて使用していく中で、使い勝手の向上や機能追加の要望が発せられることが予測される。

その場合においても、開発時と同様に各部門が連携して、真に必要な機能なのか、投資に対して十分な効果はあるのかなど、慎重に判断していくことが重要と考えている。

一方で、ユーザー部門が要望を出しやすい仕組みの構築や、いかにして効果的に実現し、不調の品質改善に努めていくかという意識も大事である。

JKK では、ステークホルダーのからの視点、働き方改革、業務の効率化の視点の双方におい

て、より一層デジタルを活用していく必要があり、本プロジェクトで得た知見を活かし、今後のシステム開発、改修につなげていきたい。