

背景1

- 現状

外部設計に指摘が多い

→外部設計の概要(要求仕様)に指摘が多い

(そもそも外部設計で要求仕様に対する指摘が入るのはおかしい)

要求仕様書を確認したが、要求仕様書は無かった(ここ哲学)

→そもそも要求仕様書はない

課題リストという形で、開発中に挙がったネタのメモが残しており、そのメモが要求仕様書代わりになっている。

- 解決したいこと

外部設計で要求仕様の指摘を減らしたい(要求仕様通りの設計を行う前に、
まず正しい要求仕様を定義したい)

→現状は重大不具合として市場流出はしていないが、要求仕様のブレは後々重大不具合に繋がる可能性が高く、今回対策する

背景2

・原因

①要求がそもそも曖昧・整理されていない

→人によって解釈が異なる状態なので、指摘が入る

②要求が仕様書として存在していない

→初めて仕様書を通じてのDRとなり、要求に関する議論で指摘が入る

③アイテムが増えすぎて、ステークホルダーの手が回っていない

→要求段階の精査が間に合わず、外部設計DRが要求確認になっており、指摘が入る

①要求がそもそも曖昧・整理されていない

- 要求獲得を体系化する

要求を5W1Hで書き出す

「いつ・どこで・誰が・何を・なぜ・どうする」

が未定義だと、何のための機能かが全く分からない

- 要求を分析・構造化する

「機能要求」「非機能要求」「制約条件」に分類する

機能要求：システムが何をするかを定義(異常発報の条件など)

非機能要求：システムがどのように動くかを定義(反応速度、機能の変更方法など)

制約条件：システムにおける前提を定義(機器構成、言語、納期など)

- 要求の合意とベースライン化

上記の要求リストをレビューし、曖昧な個所がないかを精査する

→合意を取る

②要求が仕様書として存在していない

- 要求の構造化

「要求ID/バージョン」「要求種別」「要求名称」「要求内容」「目的」「入力条件/前提条件」「出力/結果」「優先度」「検証方法」「検討結果」などを記載する

→検討漏れを防ぐ。仕様書の形にすることでレビューできる。

フォーマット化して、どの要求仕様も様式が一致していることが重要

- 要求をモデル化する

「誰が、なんの目的か(要求の外部視点)」→ユースケース図

「どのように動くか(時系列の遷移)」→ステートマシン図

「どのように実現するか(処理の流れ)」→フロー図/アクティビティ図

→誰が見ても一意に定まるように。UML(統一モデリング言語)を使うと良い

- 要求仕様書を検証する

ここまで書いておけば、さすがに査読者にも意図が伝わるはず

→「要求仕様書を作る」ことが目的ではなく、「全員の理解が一致する」ことが目的なので、はき違えないように注意

③アイテムが増えすぎて、ステークホルダーの手が回っていない

- アイテムを減らす
「ついでにやっておいて」みたいな要求は全体の品質を落とす
→優先度が低いアイテムは精査も後回しになる
- ニーズ・シーズ分析
ニーズ(顧客への価値)/シーズ(実現性)で要求の位置づけを確認

ニーズ\シーズ	高い	低い
高い	価値が高く実現しやすい	価値はあるが実現困難
低い	付加価値的要求	コストも高く価値もない

- MoSCoW法
実施するかの判断を行う
M(Must have：必須)：この要求が満たされないと仕様が成り立たない
S(Should have：できれば)：重要だけど必須ではない
C(Could have：あれば)：求められるが、余裕があれば実施
W(Won't have：いらない)：今回のスコープに含めない
→感覚でつけると形骸化する恐れあり、数値データにできると良し

まとめ

- 実際こんなに丁寧に作っている時間はない
先方も「時間が無い」とか言って付き合ってくれない
→今回の内容を完ぺきにこなすことは現実的ではない
- 最低限の要求仕様書を作成する
「ID」「目的」「条件」「結果」「検証方法」レベルで大丈夫、
エクセル1行でいいので、要求仕様一覧として見れるようにまとめる
- 要求仕様の不足はリスクとしてとらえる
キックオフ時にリスクとして定義しておく
→外部設計着手時に要求仕様が定まっていない
 - ・ リスクとして要求仕様の検討が入るバッファを設けて見積もる
 - ・ 朝礼などで周りがフォローできる状態にしておく