

要求仕様と最終製品の品質特性に関する経験的考察

中所 武司 †

大学の学科レベルで用いる会議室予約システム (Java で約 14000 行) の開発・運用の経験に基づいて、要求仕様の品質と最終製品の品質の関連について分析した。その結果、UI 駆動型の開発方法を取り、利用者と開発者の頻繁な打ち合わせが可能な状況では、インクリメンタル開発あるいはアジャイルソフトウェア開発の形態で、特に大きな問題は無かった。ただし、保守性の観点では課題が残った。

Empirical Study on Correlation between Requirements and Products

Takeshi Chusho, Meiji University

A meeting room reservation system was developed based on UI-driven approach with sufficient communication between a user and a developer. The process was similar to incremental development or agile software development. As a result, The system was released without major problems except maintainability.

1. はじめに

情報処理学会ソフトウェア工学研究会要求工学 WG では、要求仕様書の品質と最終生産物であるソフトウェアの品質との相関について検討している。本論文では、その一環として、現在、著者の所属する学科で運用している会議室予約システムの開発経験に基づいて、共通課題について考察する。

2. システムの概要

まずシステムの特徴を以下に示す。

- ・システム名: 会議室予約システム
- ・業務内容: 学科内の 4 つの会議室の予約管理
- ・利用者: 12 名の教員と 1 名の事務員
- ・システム形態: Web アプリケーション, (サーブレット, JSP, オラクル DB 利用)
- ・システム規模: Java 言語で, 155 クラス, 13700 行
- ・稼働時期: 2002 年 2 月から運用中

学科の会議室の予約は、以前は 4 つの会議室の 1 ヶ月分の予約表が紙で掲示され、手書き記入していた。本システムでは、自分の研究室や自宅の PC から 1 年先まで予約可能としたほか、大学に特徴的な機能として、授業の時限単位の予約とし、授業期間中の定期的予約も可能としている。主な機能を図 1 に示す。

利用者 (一般教員)	運用管理者 (事務員)	システム管理者 (著者)
ログイン	同左	運用管理者の
パスワード変更	同左	左記の機能
ログアウト	同左	ユーザ情報の
会議室予約	同左 (代理可)	登録・削除・
定期予約	同左 (代理可)	参照・変更
予約取り消し	同左 (代理可)	パスワード変更
予約状況照会	同左 (代理可)	会議室追加・削除
同上 (個人別)	同左 (代理可)	休日設定

図 1 会議室予約システムの主な機能

3. 開発形態

システムの開発形態の特徴を以下に示す。

- ・利用者と開発者の密なコミュニケーション
- ・UI 駆動型の分析
- ・インクリメンタル開発

第一に、著者 (教員) が利用者の立場で、学生が開発者の立場でお互いに頻繁にコミュニケーションをすることが可能な状況であった。

第二に、本システムは、Web アプリケーションであるということもあり、利用者はユーザインターフェイスの設計を通して開発者に要求を伝えることが比較的容易であった。

第三に、基本部分の開発からはじめて、実行可能な状態でプログラムの機能を付け加えていくことが可能であった。

† 明治大学 理工学部 情報科学科

結果として、今回の開発方法は、以下のようなアジャイルソフトウェア開発宣言を満たすものとなった。

プロセスとツールよりも 個人と対話を、
包括的なドキュメントよりも 動くソフトウェアを、
契約交渉よりも 顧客との協調を、
計画に従うことよりも 変化への対応を、
より尊重する。

4. 品質要求について

4.1. システムの重点品質項目

ISO の6つの品質特性の中では、機能性、信頼性、使用性が重要である。

機能性のサブ特性(合目的性、正確性、相互運用性、セキュリティ、標準適合性)の中では、特に合目的性(具体的な目標に対して適切な機能を提供する能力)が重要である。例えば、大学の教員の学科内使用に限られるので、予約は授業の時限単位とするなどである。

信頼性のサブ特性(成熟性、障害許容性、回復性、標準適合性)の中では、特に成熟性(潜在する障害によって生じる故障を回避する能力)と回復性(故障時に指定された達成水準を再確立し、データを回復する能力)が重要である。例えば、学生が開発したソフトウェアなので、1、2年後には故障が検出されても保守ができないという事情がある。

使用性のサブ特性(理解性、習得性、運用性、魅力性、標準適合性)の中では、理解性(どのように利用できるかを理解できること)も習得性(ソフトウェアの適用を利用者が習得できること)も運用性(利用者がソフトウェアの運用および運用管理できること)も重要である。利用者は一般人で、しかも専任のシステム管理者も予算もないという事情による。

4.2. システムの非重点品質項目

一方、ISO の6つの品質特性のうち、効率性(時間効率性、資源効率性、標準適合性)に関しては、スループットもレスポンスもそこそこで大丈夫とか、保守性(解析性、変更性、安定性、試験性、標準適合性)については、強い拡張要求はなく、会議室予約の基本機能だけでも有用とか、移植性(環境適応性、設置性、共存性、置換性、標準適合性)については、1サイトのみで運用という事情がある。

4.3. 要求仕様の品質項目

要求仕様の品質特性8項目については、まず正当性は重要であるが、非あいまい性と完全性と一貫性は、低コストで確認・修正が可能なので、重要性は劣る。一

方、ランク付けは、インクリメンタル開発での次の開発項目の決定に重要である。検証可能性も低コストで確認・修正を可能とするために重要である。変更可能性と追跡可能性も開発時に頻繁に拡張・変更が発生するので重要である。

5. 結果の考察

5.1. システム構成

最終的なシステム構成を図2に示す。サーブレットは53クラスで約6600ステップ、JSPは75クラスで約3700ステップ、Javaクラス(モデル)は12クラスで約1600ステップ、Javaクラス(動的HTML文書生成)は15クラスで約1800ステップ、DBテーブル(オラクル)は9個。

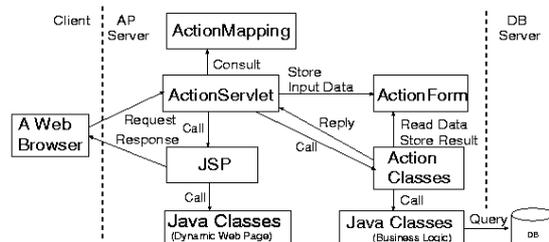


図2 システム構成(MVCアーキテクチャ)

5.2. 要求仕様の不備

正当性については、ユーザの模擬操作によるすべてのUI遷移を網羅したチェックを実施し、例外処理の洗い出しもできた。非あいまい性については、定期予約での「休日を除く」の解釈において、冬休みを含むが春・夏休みは対象外とするとか、創立記念日や大学祭の扱いが不明確であった。完全性については、定期予約に対応する取消機能の設定漏れがあった。一貫性では、同一利用者の複数ログインや運用管理者の代理操作との重複処理の可否が不明確であった。

5.3. 保守性の課題

稼働からほぼ3年になるが、バグは1件も見つからない。基本機能に絞って、機能性と信頼性と使用性を重視する方針を満たすものである。機能拡張はしなくてもよいという前提であったが、実際には、昼休みの予約化、定期予約取消、優先予約処理などの必要性が生じた。

参考文献

- [1] <http://www.chusho.jp> (著者の過去の関連文献)