

アジャイル開発におけるシステム設計

富士通株式会社
 株式会社リンクレア
 鹿島建設株式会社

田中智大
 本間峰輝
 鈴木康之

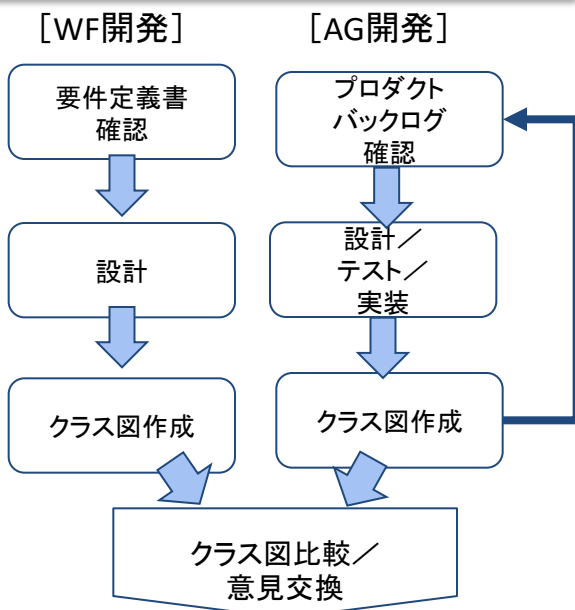
課題と目的

企業におけるアジャイル開発の採用が拡大している。全ての要件を定義・設計してから開発を行うウォーターフォール開発と比べ、インクリメンタルに設計・開発を行うアジャイル開発において気を付けるべき点を考察する。

検証方法

共通のシステム開発課題に対して全ての要件を元に設計をすすめるウォーターフォール開発とプロダクトバックログ毎に設計・開発をすすめるアジャイル開発をグループを分けて実施し、設計・開発プロセスでの気づきや作成されたクラス図を比較し、両開発における特徴を考察する

検証フロー



結果評価

■ 共通

- ・案件の特性により開発しやすい手法あり
 実装済み機能と関連が低い機能の追加型: AG開発
 実装済み機能の修正が伴う機能の向上型: WF開発
- ・開発手法・グループに関わらず、オブジェクト指向原則に違反するクラス図が散見

■ AG開発

- ・コードの可動性よりも動くものを優先すると、実装が複雑化
- ・単体要件のみに注目しすぎると、ゴールとはズレた実装になるケースも
- ・追加要件によるテスト修正負荷が高い場合は設計見直しの必要あり

■ WF開発

- ・要件定義が明確な場合設計は容易
 (ただし、開発プロジェクト問題の多くは要件定義漏れ)
- ・実装時に設計の問題に気づいた場合、後戻り工程の影響大

AG開発: アジャイル開発
 WF開発: ウォーターフォール開発

考察

以上の結果評価からアジャイル開発における設計として重要と考えるポイントを以下の通り

開発手法	ペアプログラミング, モブプログラミングで設計の属人性・低下を回避
リファクタリング	定期実施でコードのスパゲティ化を回避
ゴールの共有	認識齟齬がプロダクトにそのまま反映されるため、重要

課題

考察の深堀並びに追加観点として実施が望ましい課題は以下の通り

チーム開発評価	コミュニケーションのし易さやペアプログラミング等の実施を通し勘所を確認
システム規模影響評価	設計が複雑になると想定される大規模開発での比較確認
全体プロセス評価	要件獲得から運用まで含めた際の比較確認