Chapter13

ソフトウェア開発管理技術

# 1. 開発プロセス・手法

## 1. ソフトウェア開発手法

学習のポイント

✅ ウォータフォールモデル、プロトタイピングモデル、スパイラルモデルを中心に、特徴を覚えよう！

✅ リバースエンジニアリングの意味を覚えよう！

### １）ソフトウェア開発モデル

企業全体もしくは複数部門にまたがるような比較的大規模なシステムを基幹業務システムといいます。こうした大きなシステムの開発・設計には、システムエンジニアやプログラマと呼ばれる専門の技術者が携わります。開発期間が数ヶ月から数年と長期間になることが多いので、計画的に効率よく開発を進めていかなければなりません。ソフトウェア開発の代表的な手法には、ウォータフォールモデル、プロトタイピングモデル、スパイラルモデルがあります。

#### ①ウォータフォールモデル

ウォータフォールモデルは、開発規模が比較的大きなシステムの開発に用いられる手法です。開発工程をいくつかに分割し、決められた手順に従って開発が進められます。各工程の後戻りが難しいことから、滝の流れにたとえて「ウォータフォールモデル」と呼ばれます。そのため、ある工程の作業中、それ以前の工程にミスが発生するとその工程まで戻り、各工程をやり直す必要があります。したがって、誤りの原因が開発の初期の段階であればあるほど、修正コストは高くなります。

またこの手法では、利用者が要求を確認できるのが開発工程の終盤のテスト段階となるので、希望に添わないシステムになる危険性があります。

|  |  |  |
| --- | --- | --- |
| 開発手順 | 作業内容 | 成果物 |
| 基本計画 | 利用者の要求をまとめ、開発スケジュールを立てる。 | システム化計画書  開発計画書・要求仕様書 |
| 外部設計 | 利用者の要求に基づきシステムの機能を確定し、入力画面や帳票を設計する。 | 外部設計書  外部レビュー報告書 |
| 内部設計 | システムの機能をプログラムに分割し、プログラム間の処理の流れを決める。 | 内部設計書  内部レビュー報告書 |
| プログラム設計 | プログラムの内部構造を設計し、プログラムを構成する最小単位であるモジュールに分割する。 | プログラム設計書・  プログラムレビュー報告書 |
| プログラミング | モジュールの処理手順を設計し、プログラム言語で記述（コーディング）する。 | プログラム |
| テスト | 作成した各プログラムおよびシステム全体のテストを行う。 | テスト計画書  テスト報告書 |

ウォータフォールモデルの開発手順

|  |
| --- |
| 例題  ウォータフォールモデルによるシステム開発工程の作業内容ａ～ｆを，実施する順序で並べたものはどれか。  [作業内容]  ａ　現状の問題点を調査・分析し，対象システムへの要求を定義する。  ｂ　システムとして必要な機能をプログラムに分割し，処理の流れを明確にする。  ｃ　詳細な処理手順を設計し，コーディングする。  ｄ　テストを行う。  ｅ　各プログラム内の構造設計を行う。  ｆ　システムの要求仕様を基に，システムとして必要な機能を定義する。  ア　ａ，ｂ，ｆ，ｃ，ｅ，ｄ イ　ａ，ｆ，ｂ，ｅ，ｃ，ｄ  ウ　ａ，ｆ，ｂ，ｅ，ｄ，ｃ エ　ａ，ｆ，ｅ，ｂ，ｃ，ｄ  ウォータフォールモデルは、開発規模が比較的大きなシステムの開発に用いられるソフトウェア開発技法です。作業を、基本計画(a)、外部設計(f)、内部設計(b)、プログラム設計(e)、プログラミング(c)、テスト(d)の工程に分け、これらの工程を後戻りすることなく順に行うことを前提としています。  基本情報　平成21年度秋　問49　[出題頻度：★★★]  解答－イ |

別冊演習ドリル 》 1-938,939

#### ②プロトタイピングモデル

プロトタイピングモデルは、比較的小規模のシステム開発に用いられる手法です。システム開発の早期段階で、試作品（**プロトタイプ**）を作成します。この段階で利用者は要求を確認し評価することができるので、要求が反映された希望どおりのシステムを構築できます。

プロトタイプの作成

基本計画

外部設計

内部設計

プロトタイピングモデルの開発手順

|  |
| --- |
| 例題  プロトタイピングモデルの特徴として，適切なものはどれか。  ア　開発初期段階での試作を通して，ユーザインタフェースの確定や，応答性などの性能確認を行い，後続段階での仕様変更による手戻りのリスクを減少させる。  イ　開発プロセスを繰り返しながら改良していく成長型モデルの一種である。各繰返しで，開発コストや品質などからリスクを評価し，リスクが最小となるプロセスをとる。  ウ　短期間でシステム開発工程を一通り行って部分的に機能を完成させ，この作業を繰り返し，段階的にシステム全体を仕上げる。  エ　要求分析，システム設計，製造，テストの順に実行され，大規模システムの開発に向いている。  イ　スパイラルモデルに関する記述です。  ウ　アジャイル開発モデルに関する記述です。  エ　ウォータフォールモデルに関する記述です。  初級システムアドミニストレータ　平成20年度秋　問28　[出題頻度：★☆☆]  解答－ア |

別冊演習ドリル 》 1-940～942

#### ③スパイラルモデル

スパイラルモデルは、ウォータフォールモデルとプロトタイピングモデルの両方の考え方を取り入れた手法です。システム開発を行うときに、機能範囲を限定したシステムを作り、利用者の要求を確認しながら徐々に範囲を拡大して、最終的に１つのシステムを完成させる、という工程で進められます。開発プロセスは「設計→プログラミング→テスト→レビュー」という手順を機能ごとに繰り返し、らせん状（スパイラル）に範囲を広げながら完成へと向かいます。この手法は、システムを構成している各要素が、機能ごとに分割できるものでなければなりません。

|  |
| --- |
| 例題  要求分析から実装までの開発プロセスを繰り返しながら，システムを構築していくソフトウェア開発手法はどれか。  ア　ウォータフォールモデル イ　スパイラルモデル  ウ　プロトタイピングモデル エ　リレーショナルモデル  ア　ウォータフォールモデルは、開発工程をいくつかに分割し、決められた手順に従って開発を進めるシステム開発技法です。  ウ　プロトタイピングモデルは、開発の早い段階で試作品（プロトタイプ）を作成するシステム開発技法です。  エ　リレーショナルモデルは、データを２次元の表形式で表したデータモデルの１つです。  基本情報　平成23年度秋　問50　[出題頻度：★★☆]  解答－イ |

別冊演習ドリル 》 1-943,944

🏋プラスアルファ

**●RAD**（Rapid Application Development）

RADは、利用者の参画、少人数による開発、画面作成ツールや自動プログラミングツールなどの高度な開発ツールを用いることで、短期間でシステムを開発する技法です。ライフサイクル（開発工程）の無制限な繰返しを防ぐため、タイムボックスと呼ばれる一定の開発期間を設定します。

### ２）アジャイル

アジャイル（agile）は、短期間でシステム開発工程を一通り行って部分的に機能を完成させ、この作業を繰り返し、段階的にシステム全体を仕上げる手法です。迅速かつ適応的にソフトウェア開発を行う開発手法の総称で、代表的な方法に**XP**(eXtreme Programming：**エクストリームプログラミング**)があります。

XPは、設計よりもプログラミングやテストを重視し、ユーザ及び開発担当者間のコミュニケーションに重点を置いて、修正を繰り返しながら開発を進める技法で、少人数が担当する中小規模のプロジェクトに向いています。

XPには、開発チームが行うべきいくつかのプラクティスが定められています。代表的なプラクティスを次に示します。

#### ①ペアプログラミング

ペアプログラミングとは、開発効率を上げるために、２人一組でチームを組みプログラムコードの記述とチェックを交互に行うことです。

#### ②リファクタリング

リファクタリングとは、プログラム外部からの見た目や結果は変えずに、プログラム内部を整理し、完成度を上げるために、完成済みのプログラムコードを改良することです。

🏋プラスアルファ

**●イテレーション**

顧客の要求に柔軟に対応するために、開発単位を小さく区切って、分析、設計、実装、テストの一連の活動を繰り返し、開発を進めること

**●テスト駆動開発**

求める機能を明確化するため、テストケースを作成してテストケースをパスすることを目標にプログラミングを行うこと

**●YAGNI**（You Aren't Going to Need It）

変化に対応しやすくするために、機能を必要なものだけに絞り込むこと

**●ソースコードの共同所有**

開発者全員が改良できるように、作成者以外であってもソースコードを修正できるようにすること

**●ふりかえり**（レトロスペクティブ）

製品の完成度を上げるために、イテレーション内の事象をチーム全員で確認し改善案を考えること

**●継続的インテグレーション**

問題点や改善点を探すために、プログラミングとテストを繰り返し行うこと

**●週40時間労働**

開発効率を高めるために、残業を認めないこと

**●バーンダウンチャート**

作業の進行具合を視覚的に把握するために、縦軸に残作業量（残りの機能数やタスク数など）、横軸に時間（イテレーション、日程など）を取り、開発の初期に予定した折れ線グラフを実線で描き、その後、実績を破線の折れ線グラフで描いたもの

|  |
| --- |
| 例題  XP（Extreme Programming）において，実践することが提唱されているものはどれか。  ア　構造化設計 イ　テストツールの活用  ウ　ペアプログラミング エ　ユースケースの活用  XPは、設計よりもプログラミングやテストを重視し、ユーザ及び開発担当者間のコミュニケーションに重点を置き、修正を繰り返しながら開発を進める技法です。少人数が担当する中小規模のプロジェクトに向いています。  XPでは、短期間で開発するために、２人のプログラマがペアとなり、その場で相談したりレビューしたりしながらプログラムの開発を行うペアプログラミングを実践することが提唱されています。  基本情報　平成27年度春　問51　[出題頻度：★★★]  解答－ウ |

別冊演習ドリル 》 1-945～957

### ３）ソフトウェア再利用　🏋プラスアルファ

ソフトウェアを標準化して再利用することは、生産性を向上させるのに効果的な方法です。また、すでに品質が確保されているソフトウェアを利用するので、新規に開発する部品が減少し、品質の向上を図ることができます。しかし、部品を標準化するためには、部品として再利用することを前提に、ほかの部品との共通したインタフェースなど、使いやすさなどに留意した設計がなされていなければなりません。そのため、通常のソフトウェアを開発するよりもコストがかかります。

|  |
| --- |
| 例題　🏋プラスアルファ  ソフトウェアの再利用の説明のうち，適切なものはどれか。  ア　再利用可能な部品の開発は，同一規模の通常のソフトウェアを開発する場合よりも工数がかかる。  イ　同一機能のソフトウェアを開発するとき，一つの大きい部品を再利用するよりも，複数の小さい部品を再利用する方が，開発工数の削減効果は大きい。  ウ　部品の再利用を促進するための表彰制度などによるインセンティブの効果は，初期においては低いが，時間の経過とともに高くなる。  エ　部品を再利用したときに削減できる工数の比率は，部品の大きさに反比例する。  イ　小さい部品を再利用するよりも、大きい部品を再利用するほうが、単位規模当たりの削減効果は大きくなります。  ウ　インセンティブを与えることは、時間の経過と無関係に標準化に大いに役立ちます。  エ　再利用利益率（削減効果）は、大きな部品を再利用するほうが、小さな部品を多数利用するよりも大きくなります。  応用情報　平成24年度秋　問46　[出題頻度：★☆☆]  解答－ア |

### ４）リバースエンジニアリング

リバースエンジニアリングは、既製のソフトウェアのソースコードやオブジェクトコード、データ定義文などを詳細に分析し、設計や仕様を明らかにする手法です。設計から構築という通常の開発工程の逆をたどることから名づけられました。

他社のソフトウェア製品にリバースエンジニアリングを適用できるかどうかは、知的所有権との関連で問題となる場合があり、注意が必要です。

なお、既存のシステムを解析して設計情報を抽出し、それによって得られた情報をもとに、新たなシステムを構築する方法を**リエンジニアリング**といいます。リエンジニアリングは、現在あるシステムから設計情報を抽出するリバースエンジニアリングと、設計情報からシステムを構築する**フォワードエンジニアリング**を伴うものです。

既存システムの仕様

リバースエンジニアリング

既存ソフトウェア

追加要求

追加・修正・削除

新システムの仕様

フォワードエンジニアリング

新ソフトウェア

リエンジニアリング

|  |
| --- |
| 例題  ソフトウェアのリバースエンジニアリングの説明はどれか。  ア　開発支援ツールなどを用いて，設計情報からソースコードを自動生成する。  イ　外部から見たときの振る舞いを変えずに，ソフトウェアの内部構造を変える。  ウ　既存のソフトウェアを解析し，その仕様や構造を明らかにする。  エ　既存のソフトウェアを分析し理解した上で，ソフトウェア全体を新しく構築し直す。  既存のソフトウェアを再利用するために、原始プログラムから上位の設計情報を自動生成する方法を、リバースエンジニアリングといいます。  ア　フォワードエンジニアリングに関する記述です。  イ　リファクタリングに関する記述です。  エ　リエンジニアリングに関する記述です。  基本情報　平成29年度秋　問50　[出題頻度：★★★]  解答－ウ |

別冊演習ドリル 》 1-958～960

🏋プラスアルファ

**●マッシュアップ**

マッシュアップは、公開されているサービスを組み合わせることで、新たなサービスを提供する手法です。専門的な知識がなくても、既存のサービスを組み合わせることで、新しいサービスを短期間で開発することが可能となります。

|  |
| --- |
| 例題　🏋プラスアルファ  マッシュアップを利用してWebコンテンツを表示している例として，最も適切なものはどれか。  ア　Webブラウザにプラグインを組み込み，動画やアニメーションを表示する。  イ　地図上のカーソル移動に伴い，ページを切り替えずにスクロール表示する。  ウ　電車経路の検索結果上に，各路線会社のWebページへのリンクを表示する。  エ　店舗案内のページ上に，他のサイトが提供する地図情報を表示する。  店舗案内のページ（自分のサイトのサービス）に、ほかのサイトが提供する地図情報（他のサイトが提供するサービス）を組み合わせて新しいサービスを提供しており、既存のサービスを組み合わせて新しいサービスを提供しているのでマッシュアップに該当します。  応用情報　平成25年度秋　問50　[出題頻度：★☆☆]  解答－エ |

別冊演習ドリル 》 1-961

## 2. 開発プロセス

学習のポイント

✅ 共通フレームの問題は多数の出題実績あり！例題や演習ドリルの問題で要点を確認しよう！

### １）ソフトウェアライフサイクルプロセス（SLCP：Software Life Cycle Process）

ソフトウェアライフサイクルプロセスとは、基本計画から総合テストまでのソフトウェアの開発工程を意味します。

ソフトウェアライフサイクルプロセスの国際標準規格であるISO/IEC 12207をベースに、日本のソフトウェア産業の特性を加えて策定されたのが、共通フレームです。なお、ISO/IEC 12207をＪＩＳ規格として完全翻訳したものに**JIS X 0160**があります。

**共通フレーム**（**SLCP-JCF**：Software Life Cycle Process-Japan Common Frame）は、取引を明確化（作業内容の確認、役割分担の明確化、正確な原価の見積もり、品質の確保）し、契約上のトラブルを防ぐために、ソフトウェアライフサイクル（システムの企画、開発、運用、保守の過程）において、それらのベースとなる作業項目を１つひとつ定義し、標準化することで、発注側（利用者）と開発側双方の共通の物差しとして用いることができるように、策定されました。

共通フレームでは、システム開発に関わる技法（例えば、ウォーターフォールモデル）やツールについては、開発主体によりさまざまなものがあるため、特定の技法やツールを規定していません。さらに、ドキュメントの種類、書式等、ドキュメントの詳細も規定していません。そして、その適用に当たっては、開発モデルに合わせ作業項目の取捨選択、繰り返しの実行、複数の項目をまとめた実行などの修正（テラーリング）を行ってもよいとされています。

|  |
| --- |
| 例題  共通フレームのプロセスのうち，成果物が利用者の視点から意図された正しいものになっているかどうかを確認するプロセスはどれか。  ア　監査プロセス イ　検証プロセス  ウ　使用性向上プロセス エ　妥当性確認プロセス  ア　監査プロセスは、客観性及び独立性を保証された立場から、成果物及びプロセスが要求事項、計画及び合意に適合しているかどうかを判定するプロセスです。  イ　検証プロセスは、ソフトウェアプロジェクトが必要とするレベルに応じて、ソフトウェア製品を検証するプロセスです。  ウ　使用性向上（ユーザビリティ）プロセスは、人間の作業条件を改善し、使用者がシステムを拒否することを軽減するプロセスです。  基本情報　平成27年度秋　問50　[出題頻度：★★★]  解答－エ |

別冊演習ドリル 》 1-962～964

### ２）プロセス成熟度

組織の開発能力を評価する代表的な方法に、**CMMI**（Capability Maturity Model Integration：**プロセス成熟度モデル**）があります。

CMMIは、開発プロセスが時間の経過とともに成熟していく過程をモデル化したもので、システム開発を行う組織が開発プロセスの改善を行う際のガイドラインとなります。

その目的は、システムの開発、保守を行う組織の成熟度を評価し、向上させることによって、より良い品質のシステムを納期どおりに、より安価に調達することにあります。

CMMIでは、ソフトウェア開発組織の能力を次の５段階のレベルに分けて評価します。

|  |  |  |
| --- | --- | --- |
| レベル | 成熟度 | おもな活動 |
| １ | 管理されていない状態（初期状態） | なし |
| ２ | 類似のプロジェクトを繰り返し実施できるだけの手順や運営方針を確立した状態（管理された状態） | ソフトウェア構成管理  ソフトウェア品質保証 |
| ３ | 組織全体でソフトウェアの開発／保守の方針を策定している状態（定義された状態） | 教育プログラム  組織的プロセス定義 |
| ４ | ソフトウェアとプロセスを定量的かつリアルタイムに評価する基盤を確立した状態（定量的に管理された状態） | ソフトウェア品質管理  定量的プロセス管理 |
| ５ | 組織全体が自発的にプロセス改善に取り組んでいける状態（最適化している状態） | プロセス変更管理  技術変更管理  欠陥予防 |

プロセス成熟度と主な活動

|  |
| --- |
| 例題  CMMIの説明はどれか。  ア　ソフトウェア開発組織及びプロジェクトのプロセスの成熟度を評価するためのモデルである。  イ　ソフトウェア開発のプロセスモデルの一種である。  ウ　ソフトウェアを中心としたシステム開発及び取引のための共通フレームのことである。  エ　プロジェクトの成熟度に応じてソフトウェア開発の手順を定義したモデルである。  CMMIは、ソフトウェア開発組織およびプロジェクトのプロセス成熟度を評価するモデルで、CMMの拡張版として制定されました。プロジェクトマネジメントの成果を評価するために用いられ、CMMIを達成している企業は、プロジェクトマネジメントをきちんと実践しているとみることができます。  応用情報　平成29年度秋　問49　[出題頻度：★☆☆]  解答－ア |

別冊演習ドリル 》 1-965,966

# 2. 構成管理・変更管理

## 1. 構成管理

学習のポイント

✅ 例題や演習ドリルの問題を解きながら、要点を整理して解答力を身につけよう！

ソフトウェアには、修正や機能の追加、新システムへの移行がつきものです。したがって、マニュアルや文書も常に更新されることを前提に、いつ、どの箇所が変更されたのか管理できるようになっていなければなりません。

構成管理とは、ソフトウェア全体がどのような品目で構成されているか体系的に明らかにし、その管理方法を決めて管理することで、**ソフトウェア構成管理**（**SCM**：Software Configuration Management）とも呼ばれます。

|  |
| --- |
| 例題  ソフトウェア開発において，構成管理に**起因しない**問題はどれか。  ア　開発者が定められた改版手続に従わずにプログラムを修正したので，今まで正しく動作していたプログラムが，不正な動作をするようになった。  イ　システムテストにおいて，単体テストレベルのバグが多発して，開発が予定どおりに進捗しない。  ウ　仕様書，設計書及びプログラムの版数が対応付けられていないので，プログラム修正時にソースプログラムを解析しないと，修正すべきプログラムが特定できない。  エ　一つのプログラムから多数の派生プログラムが作られているが，派生元のプログラムの修正が全ての派生プログラムに反映されない。  システムテストにおいて、単体テストレベルのバグが多発するのは個々のプログラムの品質の問題です。  ア　定められた改版手続に従わずにプログラムを修正したために起こる不具合は構成管理に起因する問題です。  ウ　仕様書、設計書及びプログラムの版数が対応付けられていないのは構成管理に起因する問題です。  エ　派生元のプログラムの修正が全ての派生プログラムに反映されないのは構成管理に起因する問題です。  基本情報　平成28年度春　問50　[出題頻度：★★☆]  解答－イ |

別冊演習ドリル 》 1-967

🏋プラスアルファ

**●変更管理**

変更管理とは、開発のプロセスで作成されたソースコードや各種ドキュメントなどの変遷の過程（バージョンとその構成）を記録し、必要に応じて再現できるように管理することです。