

第一回ノート

ソフトウェアライフサイクル

1. **要件定義** 何を作るのか決める
2. **設計**:どのようにそれを実現するのか決める
 1. **概要設計**
 2. **詳細設計**
3. **実装**: プログラミングを行う
4. **テスト**: できあがったソフトウェアが要件や設計通りに正しく仕上がったかを確認する重要な工程
5. **運用・保守**: 実際にソフトウェアを利用、不具合や変更点を修正 昔は修正がなかったが現在はバージョンアップが中心となっている。かなりコストがかかるのでサブスクリプションが多くなっている。

この講義では設計にフォーカスを当てている

ソフトウェア開発モデル

ウォーターフォールモデル

- 基本的に後戻りすることなく実施するソフトウェア開発
- 開発するソフトウェアの仕様が確定していることが前提
- ビジネスの変化で仕様の変更があったりと不都合が生じることが多い
- 日本ではこのモデルが多い

アジャイルソフトウェア開発

- ソフトウェア開発サイクルを反復的に行う
- アメリカでは普及している

行き当たりばったりとは違うため、初学者にとって難易度が高い。実施する場合は メンターによる指導が必要

ソフトウェア品質モデル

JIS X 0129-1:2003 (ISO/IEC 9126-1: 2001) による定義。次の三つに分かれる

利用時の品質

- **有効性**: 利用者が指定された利用の状況で、正確かつ完全に、指定された目標を達成できるソフトウェア製品の能力
- **生産性**: 利用者が指定された利用の状況で、達成すべき有効性に対応して、適切な量の資源を使うことができるソフトウェア製品の能力
- **安全性**: 利用者が指定された利用の状況で、人、事業、ソフトウェア、財産又は環境への害に対して、容認できるリスクの水準を達成するためのソフトウェア製品の能力
- **満足性**: 指定された利用の状況で、利用者を満足させるソフトウェア製品の能力

内部品質・外部品質

- **機能性:** ソフトウェアが、指定された条件の下で利用されるときに、明示的及び暗示的必要性に合致する機能を提供するソフトウェア製品の能力
- **信頼性:** 指定された条件下で利用するとき、指定された達成水準を維持するソフトウェア製品の能力
- **使用性:** 指定された条件の下で利用するとき、理解、習得、利用でき、利用者にとって魅力的であるソフトウェア製品の能力
- **効率性:** 明示的な条件の下で、使用する資源の量に対比して適切な性能を提供するソフトウェア製品の能力
- **保守性:** 修正のしやすさに関するソフトウェア製品の能力
- **移植性:** ある環境から他の環境に移すためのソフトウェア製品の能力

ソフトウェア設計とソフトウェア品質の関係は、**実現したいソフトウェア品質を踏まえて、ソフトウェア設計を行う**というものである。

問題例

- 一般にソフトウェア開発はどのようなライフサイクルで行われますか？工程の一般的な名称と、それぞれの工程でどのようなことを行うのかの概要を記述してください
- 代表的な2つのソフトウェア開発の形態の名称と、それぞれの概要を記述してください。
 1. 次の品質について、JIS X 0129-1:2003での定義を記述してください
 2. 1で述べたそれぞれの品質が、最も高い優先順位で求められるようなシステム的具体例をそれぞれ挙げてください。またその理由も説明してください。