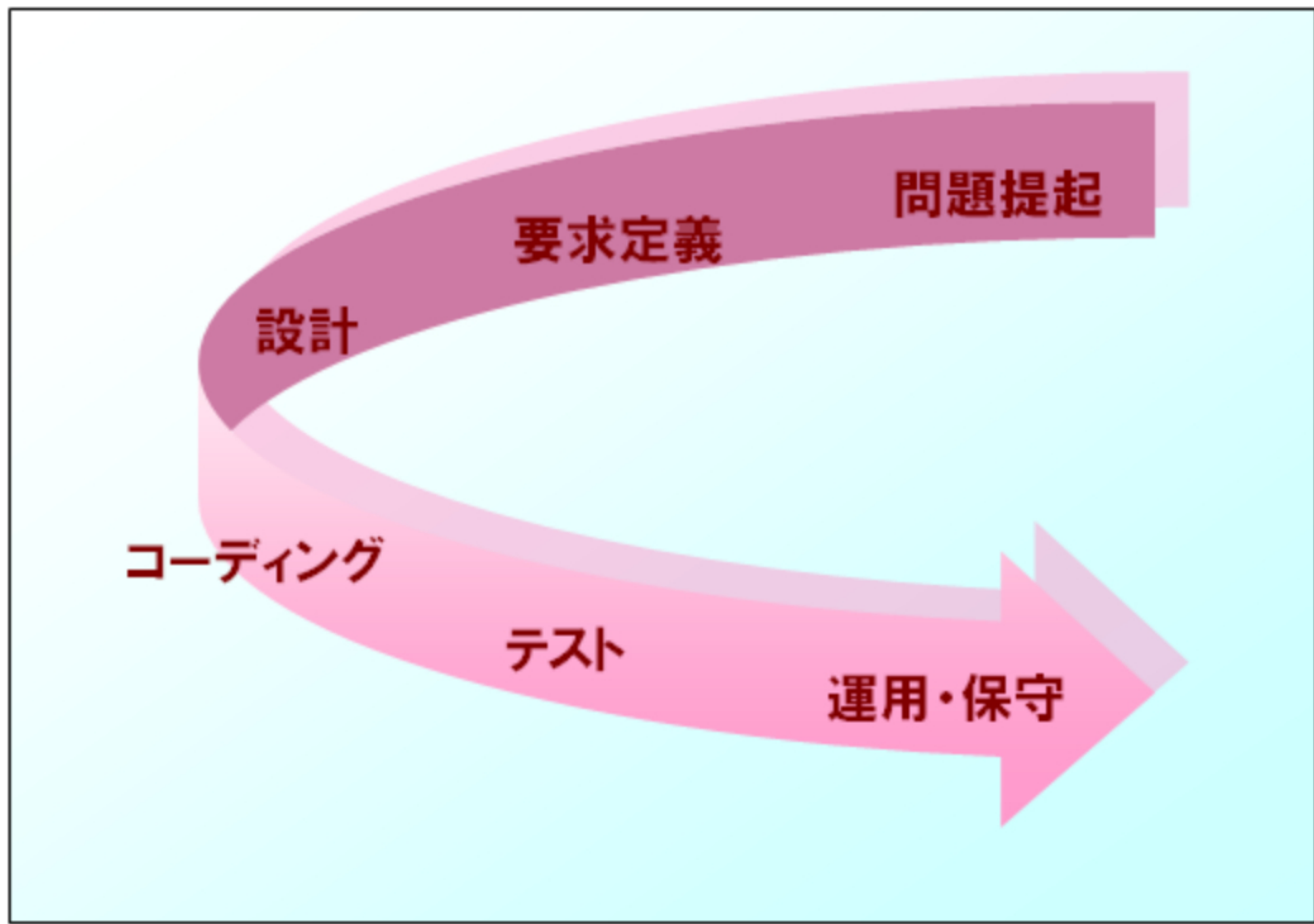
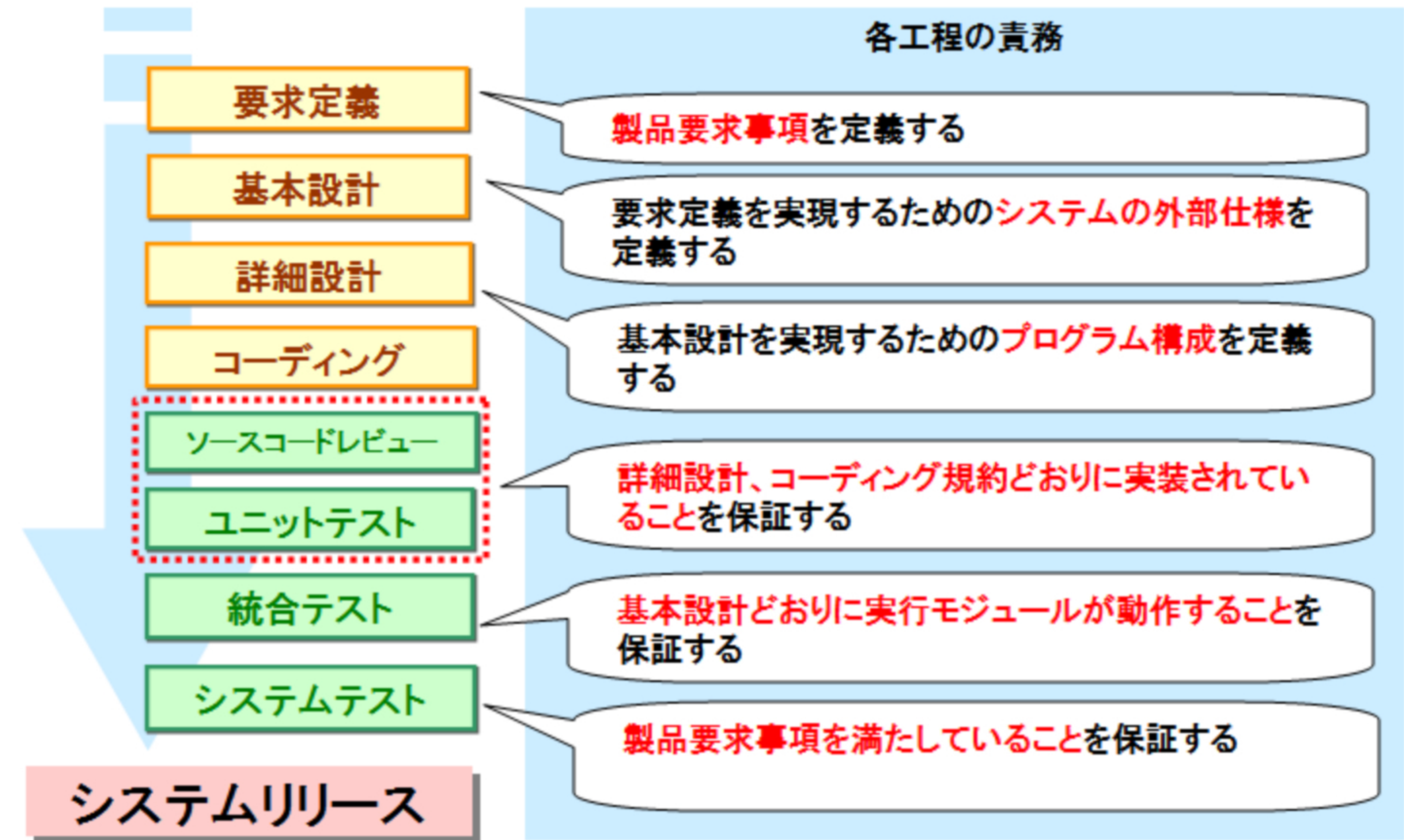


2.1. ソフトウェア開発の流れ

皆さんはソフトウェアライフサイクルという言葉をご存知でしょうか。
ソフトウェアライフサイクルとは、ソフトウェアのライフサイクル、つまり、ソフトウェア製品が産まれてから死ぬまでの生涯（≒ライフサイクル）のことです。
ソフトウェア製品を開発する私たち自身が、そのライフサイクルを知るのは当然のことですね。



要求定義からテストまでの流れが”ソフトウェアができるまでのプロセス”になります。
以下に”ソフトウェアができるまでのプロセス”の詳細を記述します。



このプロセスを構成するひとつひとつの四角が、いわゆる”工程”です。
ウォーターフォール、スパイラル、プロトタイプ…開発プロセスのモデルとして提唱されているものはいくつかありますが、どれも工程という単位でプロセスを管理します。それだけに、各工程には明確な目的と責務があるのです。

要求定義	製品要求事項をまとめる (どのようなシステムを顧客が必要としているのかまとめる) 例) 携帯電話で写真を撮りたい
基本設計	要求定義を実現するためのシステムの外部仕様を定義 (顧客が必要とする機能をいかに実現するのか決定する) 例) カメラボタンを押すとカメラモードに切り替わる
詳細設計	基本設計を実現するためのプログラム構成を定義 (基本設計をどのようにしてプログラマ的に実現するのか決定する)
コーディング	詳細設計、コーディング規約どおりに実装する
ソースコードレビュー	詳細設計、コーディング規約どおりに実装されていることを保証
ユニットテスト	詳細設計どおりに実装されていることを保証 (関数単位でテストを実施)
統合テスト	基本設計どおりに実行モジュールが動作することを保証
システムテスト	製品要求事項を満たしていることを保証

補足説明

- 工程の設定方法・呼び方は、各社・各プロジェクトによって変わります。ここで示しているのは一例です。
- ここで示しているのは、ウォーターフォール開発と呼ばれる開発プロセスです。