**プロジェクト10段階別産出物一覧**

**1. 企画・提案作業の産出物**

* **顧客インタビュー記録:**
  + **要点:** 顧客の**真の課題と必要な機能**を対話形式で記録。
  + **例:** "Aチームリーダー: '手作業データ集計に毎月3日かかり、エラーが多い。' → 自動化ダッシュボードとリアルタイムデータ連携を希望。"
* **市場調査報告書:**
  + **要点:** 市場トレンド、競合状況（長所/短所）、自社ソリューションの**差別化ポイント**を要約。
  + **例:** "競合A社: 低価格、簡単なUI / 短所: カスタマイズ不可。自社ソリューション: 柔軟なカスタマイズ機能で差別化。"
* **提案書:**
  + **要点:** 顧客の課題に対する**ソリューション、開発スケジュール、見積費用、期待効果**を提示。
  + **例:** "課題: 非効率な手作業 → ソリューション: 自動化システム。開発期間: 6ヶ月。費用: 5000万円。期待効果: 月間30時間の業務短縮。"
* **見積書:**
  + **要点:** 開発とサービス提供に必要な**項目別の詳細費用**を明記。
  + **例:** "開発人件費（上級1名/3ヶ月）: 3000万円、ライセンス: 500万円、合計: 3500万円。"

**2. 要求事項定義の産出物**

* **要求事項明細書 (SRS):**
  + **要点:** システムが\*\*「何を」すべきか、機能/非機能要件\*\*を詳細に定義。
  + **例:** "機能: ユーザーログイン（ID/PW入力 → 成功時ダッシュボードへ遷移、5回失敗時30分ロック）。非機能: 同時1000名接続時応答2秒以内。"
* **議事録:**
  + **要点:** 会議の**議論内容、決定事項、次のToDo**を記録。
  + **例:** "会議日時: 24.06.20。決定: ログイン画面UI案Aを採用。担当: キムチーム長。期限: 24.06.25。"
* **機能リスト:**
  + **要点:** システムの**全機能リストと優先順位**を表で整理。
  + **例:** "ID: F-001 / 機能: ユーザーログイン / 説明: ID/PW認証 / 優先度: 高 / ステータス: 承認済。"

**3. 基本設計の産出物**

* **システムアーキテクチャ図（テキスト表現）:**
  + **要点:** \*\*システムの全体像（構成要素と接続）\*\*をテキストで表現。
  + **例:** "ユーザー(Web) ↔️ Webサーバー(Nginx) ↔️ Appサーバー(Spring Boot) ↔️ DBサーバー(PostgreSQL)。Appサーバーは外部決済/SMS連携。"
* **データフロー図 (DFD):**
  + **要点:** システム内の\*\*データの流れ（入力、処理、保存、出力）\*\*を説明。
  + **例:** "ユーザー → [注文処理システム]（注文リクエスト）→ [在庫DB]（在庫確認）→ [決済システム]（決済リクエスト）→ [注文処理システム]（決済結果）→ ユーザー（注文確認）。"
* **システム構成図:**
  + **要点:** ハードウェア、ネットワーク、OSなど**インフラの配置**を説明。
  + **例:** "AWSクラウド: パブリックサブネット(ALB, Bastion)、プライベートサブネット(EC2 Web/Appサーバー, RDS PostgreSQL, Redisキャッシュ)。S3に静的ファイルを保存。"
* **インターフェース定義書:**
  + **要点:** モジュール/システム間の**データ送受信方法と形式**を詳細に定義。
  + **例:** "API: GET /api/users/{userId}。レスポンス: JSON {userId, userName, email}。エラー: 404 User not found。"

**4. 詳細設計の産出物**

* **DB設計書 (ERD内容含む):**
  + **要点:** **テーブル構造（カラム、型）、関係性**を詳細に定義。
  + **例:** "USER\_TB: user\_id(PK), email, password\_hash, user\_name。POST\_TB: post\_id(PK), user\_id(FK), title, content。USER\_TB ↔️ POST\_TB (1:N関係)。"
* **画面設計書 (UI/UX):**
  + **要点:** 各画面の**構成（ボタン、フィールド）、機能、画面遷移**を説明。
  + **例:** "ログイン画面: ID/PW入力フィールド、ログインボタン、「会員登録」リンク。ログイン成功時メインダッシュボードへ遷移。"
* **処理ロジック明細書:**
  + **要点:** 特定機能の\*\*具体的な処理手順（条件、順序）\*\*を定義。
  + **例:** "注文確定ロジック: ①注文照会 → ②在庫確認および引き当て（ロック適用）→ ③PG決済処理 → ④注文ステータス「決済完了」に変更 → ⑤顧客通知送信。"
* **API明細書:**
  + **要点:** **API呼び出し方法、リクエスト/レスポンス形式、パラメータ**を詳細に定義。
  + **例:** "POST /api/v1/users (会員登録)。リクエスト: {email, password, userName}。成功: 201 Created。失敗: 400（検証エラー）, 409（重複）。"
* **ユースケース明細書:**
  + **要点:** \*\*ユーザーシナリオ（ユーザーとシステムの相互作用）\*\*を詳細に記述。
  + **例:** "UC-005 商品購入: ユーザーがカートから商品を選択後決済。システムは在庫を減らし注文確定。（例外: 在庫不足時通知）。"

**5. 製造（開発）の産出物**

* **ソースコード:**
  + **要点:** 特定機能の**核となるコード断片と役割**を説明。
  + **例:** "python def encrypt\_password(password): return bcrypt.hashpw(password) : ユーザーパスワードを安全にハッシュ化する関数。"
* **コードレビュー記録:**
  + **要点:** コードレビュー時の**指摘事項、議論、解決**内容を記録。
  + **例:** "レビュアー: キム開発 / 指摘: (32行目) 'if (user.isAdmin)' → 定数利用を推奨。 / 対応: Role.ADMIN Enumに変更完了。"
* **開発ガイド文書:**
  + **要点:** **コーディング標準、環境設定、デプロイ手順**など開発ガイドライン。
  + **例:** "コーディング規約: 変数名camelCase, 関数名snake\_case。Gitコミットメッセージ: feat/fix/docsで開始。開発環境: Java 17, Spring Boot 3.0。"
* **バージョン管理ログ:**
  + **要点:** \*\*ソースコードの変更履歴（誰が、いつ、何を）\*\*を記録。
  + **例:** "commit ID: abcdef12 / コミッター: ホンギルドン / 日時: 24.06.18 / メッセージ: feat: ユーザーログイン機能実装 #123。"

**6. 単体試験の産出物**

* **ユニットテストケース:**
  + **要点:** 個別コード単位の**テストシナリオ（入力、期待結果）**。
  + **例:** "TC-U001 / 対象: calc\_sum() / 入力: (1, 2) / 期待結果: 3. / 結果: PASS。"
* **単体テスト結果報告書:**
  + **要点:** 単体テストの**総結果（パス/フェイル数）、失敗詳細**を要約。
  + **例:** "総テスト50件中48件パス、2件フェイル。失敗項目: ログイン失敗回数リセットバグ。"
* **デバッグ/バグレポート:**
  + **要点:** \*\*発見された欠陥の詳細情報（再現手順、現象、深刻度）\*\*を記録。
  + **例:** "バグID: BUG-001 / 再現: ログイン5回失敗後もログイン試行可能。 / 現象: アカウントロック機能が動作しない。 / 深刻度: Critical。"

**7. 結合試験の産出物**

* **統合テストシナリオ:**
  + **要点:** **モジュール間連携機能テスト**のシナリオ。
  + **例:** "TC-I001 / シナリオ: ユーザーログイン → 商品照会 → カート追加。 / 期待結果: 全て正常動作、データ連携確認。"
* **インターフェーステスト結果書:**
  + **要点:** インターフェース連携テスト**結果（リクエスト/レスポンスデータ、成功/失敗）**。
  + **例:** "API: /api/order (注文)。 / リクエスト: {user\_id:1, prod\_id:10}。 / レスポンス: {order\_id:100}。 / 結果: 成功。（応答時間 150ms）。"
* **欠陥レポートおよび修正履歴:**
  + **要点:** 統合テスト中に**発見された欠陥情報および修正履歴**を追跡。
  + **例:** "欠陥ID: DEF-003 / 再現: 決済モジュール連携時に断続的なタイムアウト。 / ステータス: 修正完了（コミット #abc789）。"

**8. 総合試験の産出物**

* **システム全体テスト計画書:**
  + **要点:** **システム全体の機能、性能、セキュリティテスト**計画。
  + **例:** "目標: 主要機能100%動作、同時5000名応答3秒以内。タイプ: 機能、性能、セキュリティテスト。日程: 7/1～7/15。"
* **テストケース集:**
  + **要点:** **システム全体機能検証**のためのテスト項目リスト。
  + **例:** "TC-S001 / シナリオ: ユーザー会員登録から商品購入までの全フロー。 / 期待結果: 全ステップ正常処理。"
* **性能テストレポート:**
  + **要点:** \*\*システムの性能測定（応答時間、負荷処理）\*\*結果。
  + **例:** "同時ユーザー2000名テスト: 平均応答時間1.8秒（目標2秒クリア）。最大TPS 500。CPU使用率70%。"
* **セキュリティチェックレポート:**
  + **要点:** システムの**セキュリティ脆弱性**分析結果。
  + **例:** "発見された脆弱性: XSS（クロスサイトスクリプティング）1件、SQLインジェクション0件。対応: フィルタリング強化を推奨。"

**9. 受入試験の産出物**

* **顧客受入テスト項目表 (UAT):**
  + **要点:** **顧客が直接確認する機能リスト**と結果記入欄。
  + **例:** "項目: 1.1 商品検索（キーワード「スニーカー」）/ 期待結果: スニーカーリストが正常表示。/ 顧客結果: PASS。"
* **顧客テスト結果書:**
  + **要点:** 顧客が実施した**テストの最終結果と意見**を記録。
  + **例:** "総30項目中28項目パス、2項目フェイル（バグ）。顧客意見: 「コア機能は満足だが、レポート機能の改善が必要」。"
* **承認署名／納品確認書:**
  + **要点:** **システムの最終納品と顧客承認**を公式に確認する文書。
  + **例:** "プロジェクト名: XXXシステム開発。日時: 24.07.30。上記システムを最終検収し承認しました。（顧客企業代表者署名）。"
* **ユーザーマニュアル草案:**
  + **要点:** **システムの使い方草案**（最終ユーザー向け）。
  + **例:** "[ログイン方法]: 1. ホームページにアクセス。2. 右上「ログイン」をクリック。3. ID/PWを入力後「確認」。"

**10. 運営準備／回顧の産出物**

* **オペレーションマニュアル:**
  + **要点:** システム**運用および障害対応手順**の詳細ガイド。
  + **例:** "システム起動/停止: systemctl start app.service。障害発生: ログ確認(tail -f /var/log/app.log)、担当者連絡先確認。"
* **保守計画書:**
  + **要点:** リリース後の**システム保守（障害対応、改善）計画**。
  + **例:** "保守範囲: 障害対応、月次定期アップデート。SLA: Critical障害4時間以内に復旧。"
* **プロジェクト回顧報告書:**
  + **要点:** プロジェクトの**成功/失敗要因、教訓、改善点**を分析。
  + **例:** "成功要因: 週次スクラム会議の活性化。問題点: 初期要件変更が多かった。改善: 初期要件確定プロセスの強化。"
* **FAQおよび教育資料:**
  + **要点:** **よくある質問(FAQ)とシステム教育**のための資料。
  + **例:** "Q: ログインエラーの場合どうすればいいですか？ A: ID/PWの大文字小文字確認、パスワードリセット機能を利用。[新規ユーザー向け研修] PPT資料添付。”