

第5 研究コース「人工知能とソフトウェア品質」コース（テスト分析グループ）

1. 本資料の概要

ここでは、本研究の実験で作成した資料を記載する。

2. テスト分析プロンプト

2.1. テスト分析プロンプトテンプレート

本資料は提案手法として作成したテスト分析プロンプトテンプレートであり、「#条件」部分に 2.4~2.11 章のテスト方針を追記することでテスト分析プロンプトとしている。

以降の章では、「#条件」内の「# 本箇所にテスト方針を記載する。」部分のみを記載する。

あなたはソフトウェアテストの専門家 AI です。
添付仕様書について、ソフトウェアのシステムテストの工程で実施するテスト分析を行ってください。

#定義

- ・添付仕様書は要求仕様書であり、対象となる製品・サービスに対する要求仕様が記載されている。
- ・テスト分析とは、添付仕様書を元にテストすべき対象や観点、リスクの内容などを抽出し、整理することである。

#条件

本箇所にテスト方針を記載する。

#出力項目

- ①テスト対象：テストによって確認や評価の対象となるシステム、コンポーネント、モジュールなど。
- ②テスト目的：テストを実施することで達成したい目的。例えば、欠陥の発見、品質評価、要求事項の充足確認、ステークホルダーへの根拠ある判断情報の提供など。
- ③テスト条件：テストすべき事象や機能、状態、テスト実行環境、テスト実施時の制約条件など。
- ④優先度：該当項目の優先度。最小 1~最大 5 の 5 段階で出力する。優先度の根拠も出力する。

#指示

- ・①~④の各項目にトレーサビリティを出力すること。
- ・必要に応じて表や箇条書きも利用し、抜け漏れなく網羅すること。
- ・不足情報がある場合は「不明」と明記すること。
- ・該当するものがない場合は、その旨とその理由を出力すること。
- ・該当するがテスト対象外の項目がある場合は、その項目と除外理由を記載すること
- ・AI 分析モデル情報（モデル名、バージョン、実行日時）を出力すること。
- ・テストを実施する制約や具体的な条件などの、入力ファイルに記載されている数値制約がある場合は必ずそれを含めること

#出力ファイル形式

- ・出力結果をマークダウン形式のテキストファイルで作成し、ダウンロードできるようにする。
- ・ファイル名は半角英数字を使用する。

第5 研究コース「人工知能とソフトウェア品質」コース（テスト分析グループ）

2.2. 指定なし(AI)

指定なしの場合は、2.1 章のプロンプトテンプレートは使用せず、以下のみをプロンプトとしている。

添付仕様書に対して、テスト分析を実施し、実行結果を Markdown ファイルとして標準出力してください。

2.3. 指定なし(人力)

人力でのテスト分析のためプロンプトなし。

2.4. ユーザストーリー

- ・テスト分析のために、要求仕様書に使用された製品を利用するユーザと、そのユーザが製品を使用するシナリオを想定し、そのシナリオを元にテスト項目を作成する。
 - (1) このシステム/サービスを使用するユーザを、想定されるもの全てを出力する。
 - (2) (1)のユーザが製品を使用する状況を、想定されるもの全てを出力する。
 - (3) 仕様書を元に、このシステム/サービスを使用することで実行できる事柄を、想定される限り全て出力する。「実行できる事柄」とは、仕様書に記載された機能や操作を元に実行可能な操作、動作、目標、目的などである。
 - (4) (1)の各ユーザが、(2)の状況で、(3)を実行するシナリオを網羅的に作成する。このシナリオには正常系・異常系の他に、同時/複数操作の組み合わせや通常考えづらい操作など、発生しうるあらゆるシナリオを含める。
 - (5) (4)で作成した各シナリオ毎に、テスト分析結果を出力する。

2.5. リスクベース

- ・要求仕様で定義された製品・サービスにおけるリスクを元にテスト分析を実施する。
 - (1) 製品・サービスで発生しうるリスクを、多角的な視点全て特定し出力する。
 - (2) (1)で特定したリスクに対して分析と評価を行う。分析では該当リスク発生時の深刻度と、その発生確率を想定する。またその根拠も記載する。
 - (3) (2)で設定したリスクの深刻度と発生確率を元にリスクの優先度を設定する。
 - (4) (3)で設定したリスクの優先度が高い順に、出力項目を出力する。

2.6. リスクドリブン

- ・リスクベースドテストの観点からテスト分析を実施すること。
- ・潜在的な欠陥や業務停止リスクを特定すること。
- ・リスクごとにテスト重点度・対策方針を明記すること。
- ・想定リスク一覧（原因・影響・検出可能性）を出力すること。
- ・テスト対象はリスク起点で出力すること。
- ・テスト目的はリスク低減目標を明記すること。
- ・テスト条件は重点テスト項目を出力すること。
- ・リスク分析は影響度×発生可能性マトリクスで定義すること。
- ・トレーサビリティは（リスク→テスト項目）をトレース可能にすること。

第5 研究コース「人工知能とソフトウェア品質」コース（テスト分析グループ）

2.7. 体系的手法強調

- ・ ISO/IEC/IEEE 29119 に準拠した分析手法を用いること。
- ・ 機能的観点と非機能的観点を区別して整理すること。
- ・ テスト目的はテストベースと根拠を明記すること。
- ・ テスト条件は識別子付きで体系的に出力すること。
- ・ リスク分析は影響度×発生可能性マトリクスで定義すること。
- ・ トレーサビリティは仕様項目番号との対応を出力すること。

2.8. 抽象・戦略

- ・ テスト戦略立案の観点からテスト分析を行うこと。
- ・ 機能群単位でテスト優先度を設定すること。
- ・ 品質特性（信頼性、性能、操作性など）を考慮すること。
- ・ テスト対象範囲は機能群と品質特性の対応を出力すること。
- ・ テスト目的は品質保証観点から出力すること。
- ・ テスト条件は重点テスト項目を抽出すること。
- ・ リスク分析は優先度マトリクス付きで出力すること。
- ・ トレーサビリティは品質特性→機能→テスト対応表を出力すること。

2.9. 開発時品質（主特性・副特性）

テストレベルはシステムテストを対象とする。

テスト分析はテストベース（テストの基礎となる情報源）を評価し、テストの目的を満たすために必要なテスト条件を特定するプロセスである。

テスト分析は、システム及びソフトウェアの品質特性を観点とする。品質特性には、開発時品質、使用時品質、データ品質の3つの品質があるが、今回は開発時品質を使用する。テスト分析の深さは、主特性と副特性の両方を考慮する。

テスト分析後は、テスト設計（テストケース作成など）のインプットとするために、洗い出された機能一覧、テストすべき条件（テスト条件）、およびそれらの優先度を出力すること。詳細は出力項目に示す。

テスト分析は、外部品質のみ実施する。

2.10. 開発時品質（副特性）

- ・ テスト分析は、以下に示す機能テスト・非機能テストの観点で実施する
 - 機能テスト：コンポーネント・システムが実行する機能「テスト対象が何をすべきか」を評価する。

評価では以下の品質特性を元にする。

(1) 機能適合性

- (a) 機能完全性：必要な機能を十分に提供しているか。
- (b) 機能正確性：提供する機能が期待通りに正確に動作するか。
- (c) 機能適切性：利用者のニーズに適しているか。

次ページへ続く

第5 研究コース「人工知能とソフトウェア品質」コース（テスト分析グループ）

- 非機能テスト：システムが実行する機能特性以外の属性を「システムがどのように上手く振る舞うか」を評価する

評価では以下の品質特性を元にする。

(2)性能効率性

- (a) 時間効率性：実行する機能が所定の時間内にどれだけ速く実行できるか。
- (b) 資源効率性：システムリソース（CPU、メモリ）を効果的に利用しているか。
- (c) 容量満足性：処理できるデータやユーザーの数に対する容量が満たせるか。

(3)互換性

- (a) 共存性：他のソフトウェアと共存し、干渉なく使用できるか。
- (b) 相互運用性：他のシステムやプラットフォームと効果的に連携できるか。

(4)使用性

- (a) 適切度認識性：利用者の期待に適切に応え、理解しやすいか。
- (b) 習得性：初めて使うユーザーがどれだけ早く理解して使いこなせるか。
- (c) 運用操作性：正確な操作を提供し、ユーザーが簡単に操作できるか。
- (d) ユーザエラー防止性：ユーザーの誤操作を防ぎ、誤って行った場合でも影響を最小限に抑えるか。

- (e) ユーザインタフェース快美性：ユーザインタフェースが美的で魅力的か。
- (f) アクセシビリティ：身体障害者を含む全ての人が利用しやすいか。

(5)信頼性

- (a) 成熟性：どれだけ安定しており、問題なく動作するか。
- (b) 可用性：必要な時に利用可能であるか。
- (c) 障害許容性（耐故障性）：障害やエラーに対してどれだけ耐性があるか。
- (d) 回復性：障害から迅速に回復できるか。

(6)セキュリティ

- (a) 機密性：データや情報をどれだけ機密に保護できるか。
- (b) インテグリティ：データや情報を正確に保ち、改ざんを防げるか。
- (c) 否認防止性：ユーザーが行ったアクションや取引を否認できないように保護するか。

- (d) 責任追跡性：誰が何を行ったかをトレースできるか。

- (e) 真正性：ソフトウェアが本物で信頼性があり、名前の盗用を防げるか。

(7)保守性

- (a) モジュール性：構成要素が独立しており、変更や修正が容易か。
- (b) 再利用性：他のシステムで再利用できるか。
- (c) 解析性：障害の監視やログ取得が容易か。
- (d) 修正性：変更や修正が容易か。
- (e) 試験性：試験が容易に行えるか。

(8)移植性

- (a) 適応性：異なる環境や要件に適応できるか。
- (b) 設置性：インストールが容易か。
- (c) 置換性：新しいバージョンに簡単に置き換えられるか。

(9)その他

第5 研究コース「人工知能とソフトウェア品質」コース（テスト分析グループ）

2.11. HAYST 法

テストレベルはシステムテストを対象とする。
テスト分析は、テスト分析手法として「HAYST 法」(Hierarchical Approach for Yielding Software Test conditions) を用いる。
HAYST 法に基づき、添付仕様書から次の 3 つの階層的な観点を用いてテスト条件を抽出する。
最上位（レベル 1）：要求事項（ユースケース、機能、外部インターフェースなど）
中間（レベル 2）：設計要素（内部構造、状態遷移、データモデルなど）
最下位（レベル 3）：実装および環境（エラー処理、性能要件、セキュリティ脆弱性など） テスト分析の深さは、
上記 3 階層の観点を網羅的に考慮する。
テスト分析後は、テスト設計（テストケース作成など）のインプットとするために、洗い出された機能一覧、テストすべき条件（テスト条件）、およびそれらの優先度を出力すること。詳細は出力項目に示す。

3. 採点プロンプト

あなたはソフトウェアテストの専門家 AI です。
先程実行したテスト分析結果に対して、以下の観点で採点する。

#定義

- ・添付仕様書は要求仕様書であり、対象となる製品・サービスに対する要求仕様が記載されている。
- ・テスト分析とは、添付仕様書を元にテストすべき対象や観点、リスクの内容などを抽出し、整理することである。
- ・採点項目の、各採点項目の小項目全てに対して、採点基準に従って点数を付ける。またその点数の理由も記載する。これにより算出した小項目の平均値により、各項目の点数を計算する。これを、必ず全ての採点項目に対して行う。
- ・採点結果を、html で出力する

#採点項目

①テスト対象

- ・テストによって検証すべきシステムやコンポーネント、モジュールなどが示されているか。
- ・主要機能、影響範囲が網羅されているか。
- ・テスト対象外の範囲や除外条件の明示されているか。

②テストの目的

- ・テストの主な狙い・ゴールが示されているか。
- ・目的に具体性、明確性があるか。
- ・テスト実施による期待効果が提示されているか。

#次ページへ続く。

第5 研究コース「人工知能とソフトウェア品質」コース（テスト分析グループ）

③テスト条件

- ・テストすべき機能・状態・品質特性が示されているか。
- ・テスト環境（HW/SW）が具体的に設定されているか。
- ・前提条件・開始状態が明示されているか。
- ・終了条件・テスト項目実行後の期待値が明示されているか。
- ・テストデータやシナリオが実運用可能か。
- ・テスト実施時の制約条件（時間、リソースなど）が考慮されているか。
- ・テスト時の目標値や目的状態が、仕様書に記載されている数値や状態などで具体的に明記されているか。

④優先度

- ・各テスト目的それぞれに対して優先度付けされているか。
- ・優先度の背景や根拠が入力ドキュメント内の制約条件・単語などで具体的に記載されているか。
- ・優先度と優先度の根拠の内容に整合性がとれていること。

⑤トレーサビリティ/カバレッジ

- ・テストの各項目がどの要求・要件・仕様に基づくか体系的に示されているか。
- ・要求仕様のどの要素に対応しているか明示されているか。
- ・テストケースや条件の関連要件への紐付けがされているか。
- ・テスト対象、テスト目的、テスト条件、優先度が、入力内容に対して網羅できているか。

#採点基準

- 5点：非常に良い（基準を完全に満たしている）
- 4点：良い（基準をほぼ満たしているが、軽微な改善点がある）
- 3点：普通（基準を満たしているが、改善が必要な箇所がある）
- 2点：悪い（基準を部分的に満たしているが、重要な欠陥がある）
- 1点：非常に悪い（基準をほとんど満たしていない）