

要求分析モデルのマルチプラットフォーム 適用実験に基づく十分性の分析

奥田博隆[†] 小形真平[‡] 松浦佐江子[‡]

芝浦工業大学 大学院理工学研究科 電気電子情報工学専攻[†]
芝浦工業大学 大学院工学研究科 機能制御システム専攻[‡]

1. はじめに

近年、ユーザが高機能携帯電話やラップトップ PC 等の異なる種類の端末（以下、マルチプラットフォーム）を使い分けて、サービスを利用する形態が増加している。そこで、入力方式や画面サイズ等の端末特有の特徴を生かす素早いアプリケーション開発が求められる。

我々は業務系 Web システム開発を対象に理解性向上や顧客側からの妥当性確認支援を目的として Web-UI (User Interface) プロトタイプ生成可能なモデル駆動要求分析手法を提案してきた [1]。実現するサービスは同一である為、[1]を利用したマルチプラットフォーム適用可能かつ再利用可能なモデルを構築する手法を提案した [2]。

本稿では、使用性の観点から端末の特徴が与える要求分析モデルへの影響を分析する為、1つの要求分析モデルを基に実現した Android と Web の2種類のアプリケーションの比較分析の結果を述べる。本稿の効果は、開発初期に各プラットフォームの使用性の十分性を検討する土台を与えリリース期間の短縮が可能となる事である。

2. 関連研究

要求変更が既存の機能や実行環境に与える影響を分析する手法として、谷崎らはフィーチャーモデルを用い、ソフトウェアの構成を機能と実行環境の構成要素の組み合わせに抽象化して表現し、形式化し検証する手法を提案している [3]。また、Android や Struts に代表される Web 等の様々な開発フレームワークが存在しており、要求仕様に対するフレームワークの適合性を要求仕様から「実装方法の数」, 「開発者が実装すべき処理の大きさ」等の観点で比較評価する手法 [4] が有効である。

それに対して、本研究では、画面サイズに制約を持つ Android フレームワークに固定し、その使用性の評価を UML モデルからの適用実験を通して行う。

3. 要求分析モデル

要求分析モデルは、ユースケース毎にアクタとシステムとのインタラクションに要求が顕在すると考え、振舞いとデータを数種の UML (Unified Modeling Language) モデルで表現される [1]。また、特定のフレームワークに特化した UI で解釈されるインタラクション部分と特定のフレームワーク非依存に再利用可能な部分とに分離する為、入出力データとシステムのエンティティ（永続化候補）を分離して定義する [2]。

振舞いは、主に3つのパーティションを持つアクティビティ図で定義される。ユーザの動作の起点となるトリガを含む入力項目や出力項目を定義するアクタパーティション (図 1), 入力項目検査やユーザへの通知を行う UI

インタラクションパーティション、トリガが起点となり起動する動作系列であるビジネスロジック（以降、プロセス）とそのエンティティをシステムパーティション (図 2) に別れて定義する。

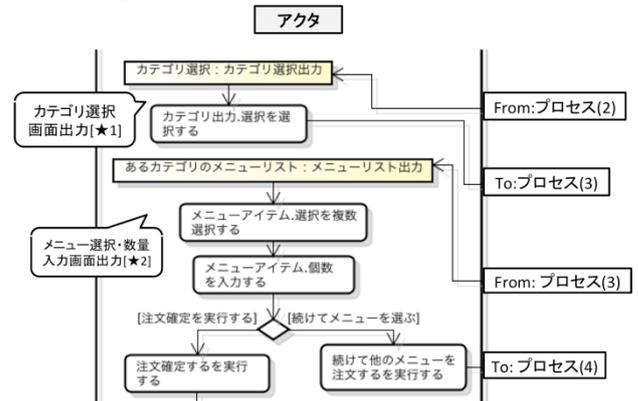


図 1: 適用事例 (アクタパーティションの一部)

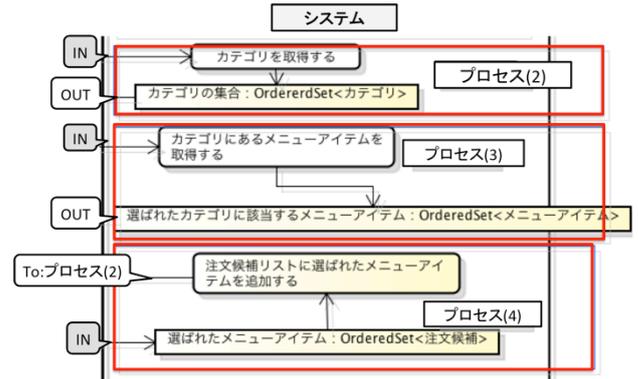


図 2: 適用事例 (システムパーティションの一部)

データ構造は UML クラス図で定義し、その具体値はオブジェクト図で定義する。

[2]では、システムパーティションの各アクションをエンティティのメソッドとして定義し、プロセス単位を統合した統括クラスを定義する。フレームワーク非依存に、データ転送用クラスを入出力データを用いて定義する。

4. 適用事例

4.1. 事例

本稿では、ある飲食店の注文を想定し顧客は Android 携帯端末から注文し、店員は PC の Web-UI から注文を行うシステムを構築した。

注文は、テーブル番号を入力し、カテゴリ選択し、メニューを選び、その数量を入力し、確認画面に飛び、注文を確定する。カテゴリ選択、メニューを選び、数量を入力する過程の要求分析モデルを図 1 と図 2 に示す。

4.2. システム構築方法

注文のフローを要求分析モデルで記述し、[2]の手法を適用する。次に[2]を適用した要求分析モデルを元に Web 及び Android に適用した。図 1 の★2 (左側の吹き出し)

Sufficiency Analysis Based on Multi-Platforms Adaptation Experiment from Requirements Analysis Model

[†]Shibaura Institute of Technology, Graduate School of Engineering and Science, Division of Electrical Engineering and Computer Science

[‡]Shibaura Institute of Technology, Graduate School of Engineering, Division of Functional Control Systems

より自動生成されたプロトタイプを元にした Web-UI による実装画面は図 3 である。



図 3: Web-UI による実装の画面

5. 考察

5.1. 入力に関する使用性向上

図 3 の様な実装を Android で行うと入力項目が過小で選択しづらく使用性が低下する為、入力項目を選択可能にし、使用性を向上する為には、画面を分割する事が解決策の 1 つである。それを実装した結果が図 5 である。

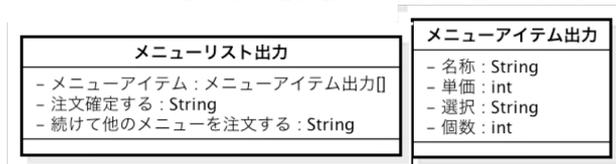


図 4: ★2 の入出力データ構造



図 5: Android による実装の画面

画面分割の為に、複数形のクラスの中で入力項目が属性として定義される箇所に注目する。該当箇所は、図 1 の★2 で、そのクラス図を図 4 に示す。メニューアイテムには「メニューアイテム. 個数を入力する」アクションと対応付く属性「個数」が存在する。そこで入力に必要なデータの観点で、属性を抽出する。メニューアイテム出力の属性からは「名称」、「単価」、「個数」、そのトリガとしてメニューリスト出力の「注文確定する」、「続けて他のメニューを注文する」を属性としてクラスを定義した (実装は図 5 右)。メニューアイテムの「名称」、「単価」を属性に持つ出力クラスとその複数形の出力クラスを定義した (実装は、図 5 左下)。モデルへの影響は入出力形式の変更の結果、プロセスに影響及ぼす事なく、フレームワーク非依存に定義したデータ転送用クラスに対して Android に特化形式を再定義した事である。

5.2. データ整理表現パターンによる使用性の向上

次に、UI パターンを整理した[5]によると Android 端末では、ドリルダウン方式 (図 5)、拡張方式 (図 6 左)、タブ方式 (図 6 右) の夫々がモデルとどのように対応するかを議論する。データ整理の表現パターンを適用できるモデルの条件は、選択するデータが関連する事から前後のプロセスに順序関係が成立し、両プロセスに

登場するエンティティ間に関連や意味的関連がある場合に適用可能である。



図 6: カテゴリからメニュー選定に至る画面

表 1: データ整理の表現のパターンまとめ

方式	データ構造の変更	プロセスへの変化	繰り返し操作性	高階層
ドリルダウン	無 (5.1 適用後)	無	×	○
拡張	有	有	○	×
タブ	有	有	○	×

3 方式とそのポイントを表 1 にまとめた。「繰り返しの操作性」とは、親・子要素を同時に閲覧可能であり、親要素を選択する事を繰り返す必要性の事である。「高階層」とは、親子要素に高い階層性への適合性の事である。拡張やタブでは同一画面内の深い階層性の表現には適さない。拡張やタブでは、既存のデータ構造に対して、同一画面内に収める為に出力データ構造をカテゴリ (カテゴリ名称) とメニューアイテム要素 (名称, 単価) を複数形の属性に持つクラスを定義した。

「プロセスへの変化」では、ドリルダウン方式では、図 2 のプロセス(2)から(3)への遷移をトリガに相当するアクション「カテゴリ出力. 選択を実行する」を伴って遷移した。拡張とタブ方式では同一画面内に表現される為プロセス(2)及び(3)がトリガなしで実行される。

6. まとめ

適用事例を通して要求分析モデルから Android や Web システムを構築し、使用性を評価した。その構築過程で発生した要求分析モデルへの影響をマルチプラットフォーム適用時の十分性の観点から考察した。

今後の課題は、データの量を変化させ使用性を評価する事やモデルパターンのメタモデルを作成し、その比較により、その十分性を検証したい。

参考文献

- [1] 小形真平, 松浦佐江子: UML 要求分析モデルからの段階的な Web UI プロトタイプ自動生成手法, コンピュータソフトウェア, Vol. 27, No. 2, pp. 14-32 (2010)
- [2] 奥田博隆, 小形真平, 松浦佐江子: ビジネスロジックモデリングのアプリケーション開発フレームワークへの依存性分析, 信学技報, Vol. 111, No. 211, pp. 61-66 (2011)
- [3] 谷崎裕明, 片山卓也: ソフトウェアシステムの構成変更における整合性判定フレームワークの提案, コンピュータソフトウェア, Vol. 28, No. 2, pp. 217-235 (2011)
- [4] 善明晃由, 小林隆志, 佐伯元司: フレームワークの要求仕様に対する適合性の評価手法, 信学技報, Vol. 103, No. 481, pp. 7-12 (2003)
- [5] Android Interaction Design Patterns: <http://www.androidpatterns.com/>