

特定問題領域に特化した Android アプリケーション 自動生成ツールの構築と適用評価

飯山 大樹[†] 中所 武司[‡]

[†] 明治大学大学院 理工学研究科 ソフトウェア工学研究室

E-mail: [†] hirokiiiyama2040@gmail.com, [‡] chusho@cs.meiji.ac.jp

あらまし 近年, PC やスマートデバイスは, 高機能化及び低価格化により広く普及している. それに伴い, Web アプリケーションやスマートデバイス向けアプリケーションを活用し, 日常の業務処理を行う人々 (エンドユーザ) も増加している. しかし, 小規模なアプリケーションを情報技術の専門家に構築してもらうのは, コスト的に見合わないケースが多い. 小規模な業務であれば, エンドユーザ自らが自分の業務に合ったアプリケーションを設計し作成することが望ましい. 本研究では, リユースサービスに特化した Android アプリケーションを対象に, エンドユーザ自身が Android アプリケーションの仕様を定義することで Android アプリケーションのソースコードを自動生成するビジュアルツールを開発した. また, そのツールの適用実験により, エンドユーザによる Android アプリケーション開発が可能であることを確認した.

キーワード エンドユーザ, Android, リユースサービス, 自動生成

Domain Specific Android Application Supporting Tool

Hiroki Iiyama[†] Takeshi Chusho[‡]

[†] Software Engineering Laboratory, Graduate School of Science and Technology, Meiji University

E-mail: [†] hirokiiiyama2040@gmail.com, [‡] chusho@cs.meiji.ac.jp

Abstract In recent years, the number of PCs and smart device is on the increase. And the number of end-users who use Web applications and applications for smart device to do their daily works is also on the increase. However, it is difficult for end-users to make orders for small-scale applications to IT professionals, because of the cost. In such a case, it is a good way that an end-user makes his own small-scale application by himself. We developed a visual tool generating the Android application code for reuse service automatically. This tool uses an Android application model which should be defined by an end-user. The effectiveness of this method is confirmed by feasibility studies.

Keywords End-user, Android, Reuse service, Automatic generation

1. はじめに

近年の PC やスマートデバイスの高機能化及び低価格化, インターネットの普及により, インターネットを利用する PC やスマートデバイスは日々の生活やビジネスにおいて広く普及している. それに伴い, ネットワークを通して Web アプリケーションやスマートデバイス向けのアプリケーションを活用し, 日常の業務処理を行う人々 (エンドユーザ) も増加している.

ところが, 小規模なアプリケーションを情報技術の専門家に構築してもらうのは, コストの問題から敬遠しがちである. 小規模な業務であれば, エンドユーザ自らが自身の業務に合ったアプリケーションを設計し構築することが望ましい.

しかし, エンドユーザがアプリケーションを構築するには多くの知識やスキルが必要であり, アプリケーション構築の支援が非常に重要である.

本研究では, エンドユーザによる小規模な Android アプリケーション構築支援として, Android アプリケーションのソースコードを自動生成するビジュアルツールを構築した. アプリケーションの対象業務として, リユースサービスを選定している. また, そのツールを用いて Android アプリケーションが開発できることを確認し, 適用評価を行った.

2. エンドユーザ主導開発の課題と解決策

1つのアプリケーションを構築するためには, アプリケーション開発者に高いスキルが要求され, スキルの取得は容易ではない. アプリケーションを設計するためには, 以下の3つが重要である [1][3].

- ・ ユーザインターフェース設計
- ・ データベース設計
- ・ ビジネスロジック設計

ユーザインターフェース設計では, Android アプリ

ケーションの場合、ビジュアルにレイアウト設計が可能なツールが存在するため、本研究の対象外とする。データベース設計では、一般的にはテーブル構造の理解や SQL 文の理解などが必要であるが、エンドユーザ主導開発にとって大きな障害になることは考えにくい。ビジネスロジック設計では、エンドユーザが携わっている業務のビジネスロジックは様々なので、それを正しく記述することは容易ではない。

そこで、エンドユーザ主導 Android アプリケーション構築支援のために、ソースコードを自動生成するビジュアルツールを構築した。データベース設計とビジネスロジック設計に対応するソースコードの生成が中心となる。

3. 例題アプリケーション試作

3.1 例題アプリケーション選定

例題として小規模な不用品再利用アプリケーションを選定した。近年、情報技術を応用して持続可能な社会のための環境保護に貢献する「Green-by-IT」が期待されている。不用品再利用サービスに関して、もし、担当者自身がアプリケーションを構築することができれば、扱える品物の量や利用者の数は格段に増加すると思われる。この不用品再利用アプリケーションを明治大学情報科学科に所属する人向けに構築した。

3.2 利用シナリオ

例題アプリケーションの利用シナリオを図 1 に示す。

- ① 不用品提供者は、所有する不用品に関する情報を入力し、登録する。
- ② 不用品希望者は、システムに登録されている不用品の中で、引取を希望する不用品に対して申請する。
- ③ システムは、申請された不用品の不用品提供者に、メールを送信する。
- ④ 不用品提供者は、その申請に対して承諾・拒否を行う。
- ⑤ 承諾の場合、不用品希望者と不用品提供者同士で不用品の受け渡しを行う。

3.3 機能と画面遷移図

図 2 に画面遷移図を示す。構築した例題アプリケーションは以下の機能を持つ。

- A) 自動ログイン機能
- B) 新規登録機能
- C) 不用品の登録、編集、削除機能
- D) ユーザ情報編集機能
- E) 不用品申請機能
- F) 不用品申請取消機能
- G) 不用品承諾・拒否機能
- H) メール送信機能

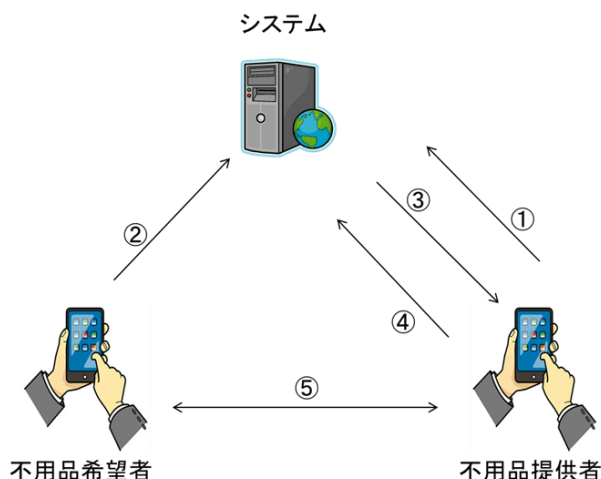


図 1 利用シナリオ

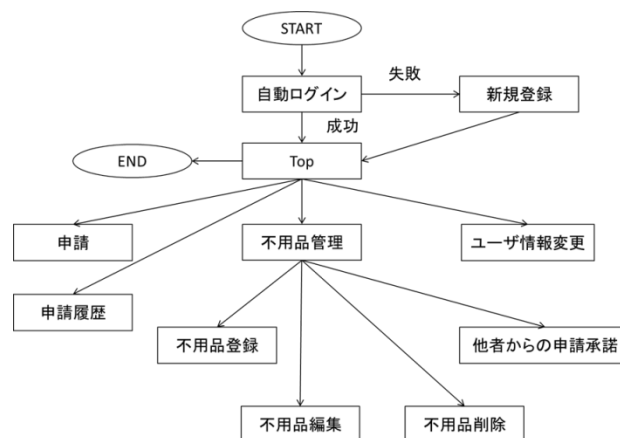


図 2 画面遷移図

3.4 サーバ側のアプリケーション構築

例題アプリケーションは、利用端末から外部サーバにアクセスしたデータの永続保存やメール送信を行う必要がある。エンドユーザ主導開発の観点から、クラウドサービスである Google App Engine を利用している。Google App Engine 上に以下の機能を持つ Web アプリケーションを構築した。

- ・ Google App Engine のデータストアにアクセスし、CRUD (Create, Read, Update, Delete) 処理を行う。
- ・ Google が提供する Mail Java API を利用し、メール送信を行う。

なお、例題アプリケーションでは、当初、スマートデバイスの特長を生かした機能を実現することを想定していた。すなわち、マッチングサービスに有用な機能として、Android 端末のカメラ機能を利用した不用品の画像登録作業の簡易化を実現する予定でいたが、開発期間の都合により、今回の試作では未実装である。

4. フレームワーク構築

4.1 機能

例題アプリケーションのソースコードを整理し、業務固有部分と共通部分に分離してそれぞれ集約した。共通部分に関しては、Android アプリケーションを構築する際に再利用可能な部分になるので、共通部分のファイル群をリユースサービスに特化したフレームワークとする。

本フレームワークは、簡易な記述で以下の機能を実現できる。

- ① Google App Engine のデータストアにアクセスし、CRUD 処理を行う。
- ② 指定のメールアドレスにメールを送信する。
- ③ マッチングテンプレート機能を利用して、マッチングサービス特有の機能を実現できる。

4.2 マッチングテンプレート機能について

本フレームワークのマッチングテンプレート機能を利用することで、以下の3種類のマッチング方法を簡易な記述で実現できる。

マッチング方法 1～承諾～

不用品希望者 A が不用品 C を申請した場合、不用品提供者 B はその申請情報 D に対して承諾・拒否する必要がある。例題アプリケーションではこの方法を採用しており、申請処理、申請取消処理、申請承諾処理、申請拒否処理の4つの処理を利用できる。

マッチング方法 2～ランダム～

複数の不用品希望者が不用品 C を申請した場合、承諾する不用品希望者をシステム側がランダムに決定する。そのタイミングは、不用品提供者が決定でき、申請処理、申請取消処理、申請決定処理の3つの処理を利用できる。

マッチング方法 3～先着順～

不用品希望者が申請した瞬間に不用品の受け渡しを決定する。決定した後でも申請を取り消すことは可能とする。申請処理と申請取消処理の2つの処理を利用できる。

5. エンドユーザ主導ビジネスロジック定義方法

5.1 イベント駆動型プログラミング

Android アプリケーションはイベント駆動型のプログラミングなので、イベント（ユーザ動作）が発生した時、どのようなイベントハンドラ（どのような処理を実行させるか）にするかを決定する過程がプログラミングである。

5.2 コンポーネントを利用した定義方法

イベントハンドラを考える場合、どのような条件でどのような処理を実行させるかを定義することが重要

である。つまり、「条件1から真偽を判断し、真の場合処理A、偽の場合処理Bを実行する」となる条件1、処理A、処理Bをエンドユーザが定義する。また、条件を複数考えるケースや、処理を複数実行する可能性もある。これらの概念図を図3に示す。

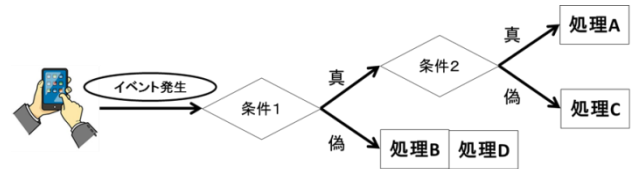


図 3 エンドユーザ定義方法の概念図

図3のような条件や処理を詳しく定義することはエンドユーザには難しいため、条件コンポーネントと処理コンポーネントを用意する。これらのコンポーネントを正しく組み合わせて利用することにより、ビジネスロジックを実現する。条件コンポーネントと処理コンポーネントを利用した定義方法の概念図を図4に示す。

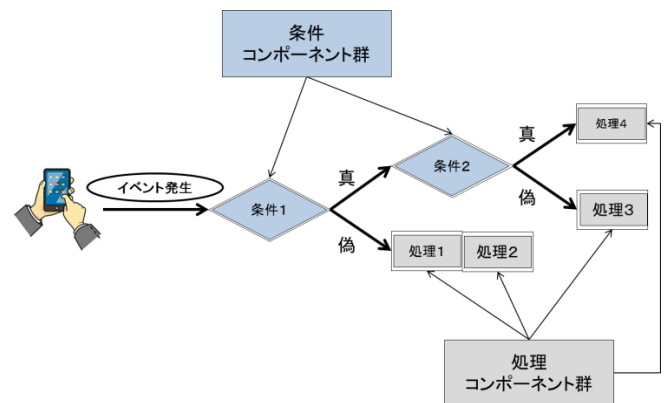


図 4 コンポーネントを利用したビジネスロジック定義方法の概念図

5. 自動生成ツールの構築

5.1 支援範囲

スマートフォン上でAndroidアプリケーションを利用するには、AndroidアプリケーションのAPKファイルが必要になる。図5に、本研究で構築した自動生成ツールを利用した場合の、APKファイル生成までの流れを示す。

エンドユーザは、本ツールの表示内容に沿って、Androidアプリケーション構築に必要な情報を入力する。本ツールは、入力された情報を基にソースコードを生成し、予め用意されたAndroidアプリケーションのプロジェクトフォルダに格納する。フォルダの内容は、Androidアプリケーション開発環境が整ったeclipseのAPKファイル生成機能を利用して、容易にAPKフ

ファイルに変換することができる。生成した APK ファイルをスマートフォンに保存してインストールすることで、構築した Android アプリケーションを利用することができる。

図 5 の赤枠で囲んだ部分が、本ツールの Android アプリケーション開発支援範囲である。

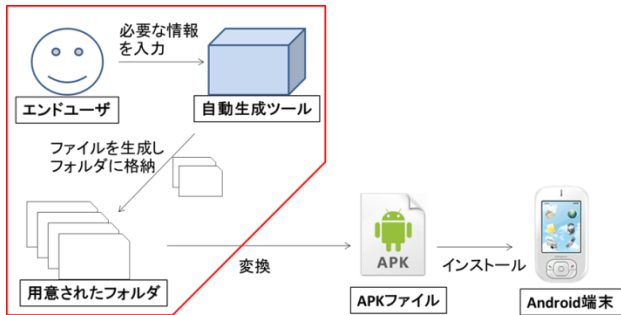


図 5 本ツールの開発支援範囲

5.2 各設計の流れとファイル生成部分

一般的なシステム開発では、UI 設計、BL 設計、DB 設計に分離できる。Android アプリケーション開発の UI 設計では、記述したファイルは res/layout フォルダ、res/values フォルダに格納する。UI 設計は本研究の対象外にしているため、本ツールではこのフォルダにファイルを生じない。よって、エンドユーザには、他のビジュアルツール等を用いてレイアウトファイルを作成し、フォルダに格納する必要がある。レイアウトを作成することは本ツールでは行わないが、レイアウトで利用するコンポーネントの情報を BL 設計で利用するため、コンポーネント情報を本ツールに入力する必要がある。

DB 設計内容は、独自に作成した src/object フォルダに格納される。通常、作成する Java ファイルは src 内ならどの場所でも構わないが、構築したフレームワークに対応するために src 内の object フォルダにファイルを格納する。

BL 設計内容は、src フォルダに格納する。生成するファイルは Activity の役割を持つ Java ファイルなどである。イベントリスナーも Activity ファイルに記述される。

また、UI 設計、DB 設計、BL 設計以外にアプリケーション構築に必要な情報があり、このアプリケーション情報を、入力する必要がある。

本ツールを用いた Android アプリケーション構築の流れとしては、まず他のツールを用いて UI 設計を行う。エンドユーザがアプリケーションを構築する場合、まず UI 設計を行うことでアプリケーションがイメージしやすくなるためである。次に、本ツールを用いて

アプリケーション情報を入力する。次に DB 設計を行う。UI 設計、DB 設計を終えることで、アプリケーションの全体像がエンドユーザにも明確になると考える。その後、コンポーネント定義を行う。コンポーネント定義情報は、Activity ファイルで利用される。UI 設計と DB 設計で定義した内容を利用する必要があるため、このタイミングで行う。最後に BL 設計を行う。

各設計とソースコード生成部分の関係を図 6 に示す。赤枠で囲んだ部分が src フォルダと res フォルダであり、各設計で生成されるファイルを矢印で示す。

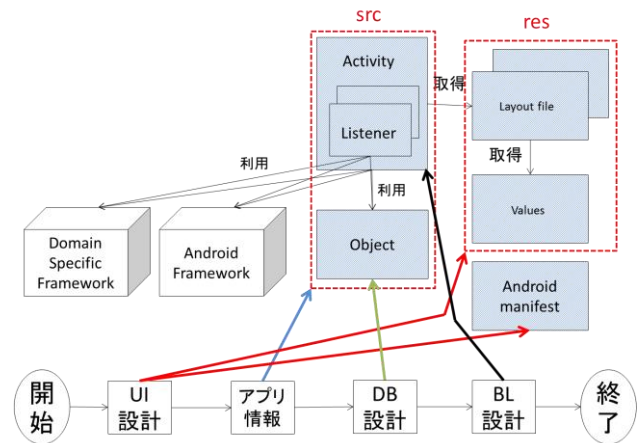


図 6 各設計とソースコード生成部分の関係

5.3 アプリ情報を入力

アプリ情報入力とは、UI 設計、DB 設計、BL 設計のどの部分にも該当しないが、Android アプリケーションを構築する上で必要となる情報の入力である。入力内容は、Google App Engine に接続するための URL である。エンドユーザは、本ツールに設定した URL を入力する。

5.4 UI 設計情報の入力

UI 設計情報の入力とは、レイアウトで利用するコンポーネントの種類を選択し、コンポーネントに定義した ID の入力を行うことである。

他のビジュアルツール等を用いて生成した XML ファイルには、GUI コンポーネントがどのように画面上に配置されるのかが記述されている。これらの GUI コンポーネントの種類と定義した ID を本ツールに入力し、その情報を参照して BL 設計を行う。

5.5 データベース定義

本ツールを用いたデータベース定義では、構築した本フレームワークを最大限利用するため、カインドとプロパティが予め一部定義済みとしている。

エンドユーザは、自身の業務に合わせて、各カインドに追加するプロパティを定義すればよい。

次に予め定義されているプロパティと新たに定義

したプロパティに対して、バリデーションの設定を行う。本ツールで予め用意されているバリデーションを選択して、各プロパティに適切に設定する必要がある。

5.6 コンポーネント定義

ビジネスロジック設計の前に、UI 設計で定義したコンポーネントに対して、タグ定義や表示する情報を定義する必要がある。GUI コンポーネントに関することなので、意味的には、UI 設計に近いと思われる。

レイアウトで利用したテキストボックスなどのコンポーネントに対して、どのようなデータを持つのかについて、プロパティ名を指定して定義する。また、リストなど動的に表示内容が変化するコンポーネントに対して、表示する内容を定義する。

5.7 ビジネスロジック定義

ビジネスロジック定義では、本ツールが用意する値コンポーネント、処理コンポーネント、条件コンポーネントを組み合わせることで実現する。

値コンポーネント

値コンポーネントは、処理コンポーネントや条件コンポーネントで利用する部品である。値コンポーネントの種類に関しては、例題アプリケーション構築や、食品管理 Android アプリケーション構築[4]時に利用した値から選定している。現段階では、11種類の値コンポーネントを用意しているが、より多様なアプリケーションを本ツールで構築可能にする場合は、これらの値コンポーネントをより充実させる必要がある。

条件コンポーネント

条件コンポーネントは、設定された条件から必ず真と偽を決定するものである。プログラミングでは if, else 文に該当する。表 1 に示す機能を用意している。マッチング用の条件コンポーネントも用意しており、マッチング用条件コンポーネントでは、マッチングサービス特有のルールを容易に設定することができる。

表 1 条件コンポーネント一覧

種類	説明
条件なし	条件がない場合に利用する。
2つの値を比較	2つの値を比較する。
バリデーション処理	入力された内容に対してバリデーションを行う。
DB 値の比較	エンティティの検索を行い、そのエンティティが存在するのか、あるいはいくつ存在するか判定する。
申請条件	設定された申請ルールに関して、そのルールを満たしているかを判断し、申請可能な場合は真、申請不可の場合は偽となる。

申請取消条件	設定された申請取消ルールに関して、そのルールを満たしているかを判断し、申請取消可能な場合は真、申請取消不可の場合は偽となる。
申請承諾・拒否条件	設定された申請承諾ルールに関して、そのルールを満たしているかを判断し、申請承諾可能な場合は真、申請承諾不可の場合は偽となる。

処理コンポーネント

処理コンポーネントは、条件コンポーネントの真偽に対応する処理の定義に用いるもので、表 2 に示す種類を用意している。それぞれの真偽に対して複数の処理コンポーネントを定義することが可能である。

表 2 処理コンポーネント一覧

種類	説明
未選択	処理を実行しない場合。
条件	さらに条件を付け加えたい場合。
画面遷移	現在の画面から別の画面に遷移する処理。
DB 操作	DB の Create (登録), Update (更新), Delete (削除) を行う処理。
ダイアログ	ダイアログを表示する処理。
トースト	トーストを表示する処理。
端末データ保存	プリファレンス機能を利用して端末に値を保存する処理。
動的ラベル表示	値を動的に画面に表示する処理。
システムからメール送信	Google App Engine のメール送信サービスを利用してメールを送信する処理。

また、表 2 以外にも、各マッチング方法に応じた特別な処理を用意しており、マッチング特有の処理を容易に実現できる。

ビジネスロジック定義方法の概念図を図 7 に示す。各コンポーネントを適切に組み合わせることにより、各イベントに対するイベントリスナーを定義する形となる。エンドユーザは、図 8 のような本ツールのビジネスロジック定義画面からコンポーネントを選択することで定義する。

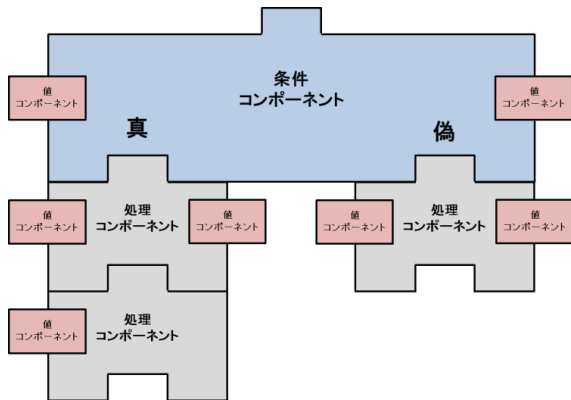


図 7 ビジネスロジック定義方法の概念図

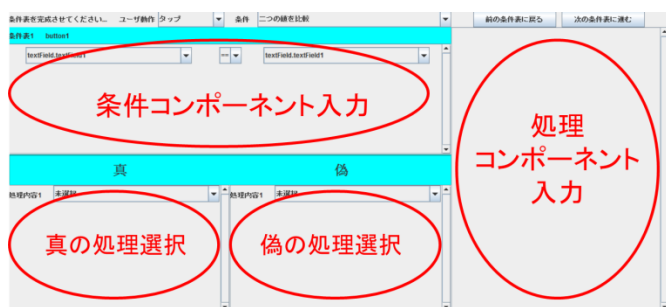


図 8 ビジネスロジック定義画面

6. 適用評価

6.1 構築実験

本ツールを用いて、不用品再利用アプリケーションの構築実験を行った。構築したアプリケーションは例題アプリケーションとほぼ同様のものである。実験参加者は以下の3名である。

本ツールの開発者

指定の Android アプリケーションが構築可能かチェックを行うため筆者自身による構築実験を実施した。

IT知識を持つ本研究室に所属する学生

本ツールに関する知識はないが、アプリケーション開発の経験がある本研究室の院生に構築実験を依頼した。

IT知識を持たない社会人

本ツールに関する知識はなく、アプリケーション開発の経験も一切ない、社会人の友人に構築実験を依頼した。

構築に必要とした時間を表3に示す。

表 3 構築に必要とした時間

構築実験参加者	必要とした時間
本ツールの開発者	1時間10分
IT知識を持つ学生	1時間50分
IT知識を持たない社会人	—

本ツール開発者は、操作方法などを理解しているの

で、IT知識を持つ学生より早い時間で構築できた。

IT知識を持つ本研究室の学生は、構築時間は執筆者よりも多いが、操作方法と考え方に慣れてくると、執筆者と同等のスピードで設計できるようになった。本ツール利用中は、IT知識を有するので、コミュニケーションを容易に行えた。

IT知識を持たない社会人に構築実験を依頼したが、構築するまでに至らなかった。開始20分ほどで構築は不可能であると実験参加者と執筆者の意見が一致し、構築を断念した。リストに表示する内容の定義方法など、初見では理解が難しい部分があった。

6.2 評価

本ツールを用いて、Androidアプリケーションのソースコードを記述したファイルを生成できることを確認した。生成したファイルを利用してAndroidアプリケーションを構築し、動作することを確認した。数時間で、小規模な不用品再利用Androidアプリケーションを構築可能であることを確認した。

本ツール利用中の質問数の多さや、IT知識の有さない実験参加者が構築までに至らなかった点から、本ツールの利用に際しては、利用方法を詳細に解説したマニュアル書の準備や、本ツールに関する事前研修の実施が必要であることがわかった。エンドユーザが本ツールを利用してアプリケーションを開発するためには、エンドユーザのIT関連の予備知識の有無に応じた対応が必要である。

7. 終わりに

本研究では、例題として不用品再利用Androidアプリケーション選択し、Google App Engineを活用してそのアプリケーションを構築するとともに、その結果に基づいて、マッチングサービスに特化したフレームワークを構築した。

さらに、エンドユーザ主導開発支援として、Androidアプリケーションのソースコードを自動生成するビジュアルツールを構築し、適用評価を行った。

文 献

- [1] 中所 武司：“マッチングシステムを例題としたエンドユーザ主導開発方式に関する考察” 信学技報 vol.114, no.292, KBSE2014-28, pp.1-6, 2014.11.
- [2] 李 静：“エンドユーザ向け Web アプリケーションフレームワークの提案と評価” 明治大学大学院理工学研究科 2012年度修士学位請求論文
- [3] 橋本 竜太：“エンドユーザ主導 Web アプリケーション開発技法の提案と評価” 明治大学大学院理工学研究科 2013年度修士学位請求論文
- [4] 飯山 大樹：“Android アプリによる食品管理システムの構築と運用の考察” 明治大学情報科学科 2013年度卒業論文