

iHAC Recipe Creator: iHACシステムにおける 機器連携レシピを作成するモバイルアプリケーション

近藤 浩之^{†1} 林 宏輔^{†2} 鈴木 秀和^{†2}

^{†1} 名城大学理工学部 ^{†2} 名城大学大学院理工学研究科

1 はじめに

家電連携の制御内容が記載されたレシピに基づいて機器連携を可能とする宅内用デバイス iHAC (intuitive Home Appliance Control) Hub [2] が提案されている。

本稿では機器連携が必要となるレシピを作成するためのモバイルアプリケーション iHAC Recipe Creator を提案する。

2 iHAC Hub の概要

これまでユーザが部屋名や機器の動作条件、動作内容などの情報が記述された JSON (JavaScript Object Notation) 形式のレシピを実行することにより、異種規格の機器連携を可能とする iHAC システム [1] が提案されてきた。iHAC Hub とは、iHAC システムを応用し、温度や湿度などの環境情報に基づいた機器連携を実現する宅内用デバイスである。発生する事象、ルールが発火条件、実行される操作を ECA ルールに基づいて記述したレシピを定義し、レシピを iHAC Hub 内で解析する。レシピの動作条件を満たした場合、機器制御を行う通信処理部を呼び出すことで環境情報に基づいた機器連携が可能になる。しかし、既存の iHAC Hub ではあらかじめ作成されたレシピを用いた機器連携のみ可能であり、レシピの作成方法については検討されていない。

3 提案アプリケーション

3.1 構成

iHAC Recipe Creator はユーザがレシピ作成に必要な情報をタッチ操作することでレシピを作成できるアプリケーションである。図 1 にアプリケーションの概要を示す。iHAC Recipe Creator は UI、レシピ作成 API、レシピ送信 API で構成されており、iHAC Hub はレシピ受信 API、既存機能であるレシピエンジンで構成されている。UI はユーザのタッチ操作により、レシピ実行のきっかけとなる種類を示すレシピタイプやレシピの動作条件、連

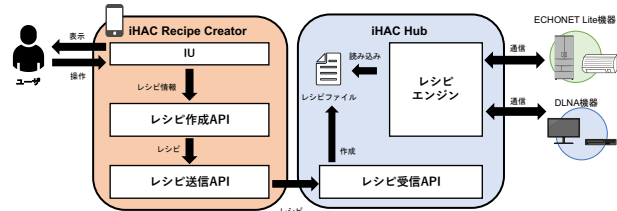


図 1 提案アプリケーションの概要

```
{
  "recipeType": 1,
  "event": [
    {
      "number": 1,
      "triggerDeviceId": 1
    }
  ],
  "condition": [
    {
      "number": 1,
      "triggerDeviceStatus": "Power",
      "triggerDeviceStatusValue": "On"
    }
  ],
  "action": [
    {
      "controlDeviceId": 4,
      "controlMethod": "ENL",
      "controlDeviceStatus": "Power",
      "controlDeviceStatusValue": "On"
    },
    {
      "controlDeviceId": 4,
      "controlMethod": "ENL",
      "controlDeviceStatus": "Color",
      "controlDeviceStatusValue": "lampColor"
    }
  ]
}
```

図 2 ECA ルールに基づいて記述したレシピの例

携する機器の動作内容などのレシピ作成に必要なレシピデータを選択する。レシピ作成 API はレシピデータをもとにレシピを作成する。レシピ送信 API は作成されたレシピを iHAC Hub に送信する。レシピ受信 API は iHAC Recipe Creator からレシピを受信し、JSON ファイルとしてレシピを出力する。

家電の連携動作を記述したレシピは ECA ルールに基づいて記述し、JSON 形式で作成する。ECA ルールとはアクティブデータベースにおいて自動的に実行する処理を定義するために用いられるルールであり、ルール実行のトリガとなる Event、ルール実行の条件 Condition、実行される処理 Action で構成される。例えば、「リビングのエアコンの電源が ON になったとき、リビングの照明の電源を ON にして光の色を電球色にする。」というレシピは図 2 のように記述される。

iHAC Recipe Creator: A Mobile Application for Creating Device Cooperation Recipe in iHAC System

Hiroyuki Kondo^{†1}, Kosuke Hayashi^{†2} and Hidekazu Suzuki^{†2}

^{†1} Faculty of Science and Technology, Meijo University

^{†2} Graduate School of Science and Technology, Meijo University

3.2 動作

3.2.1 iHAC Recipe Creator

レシピ作成およびレシピ送信の流れについて説明する。ユーザは iHAC Recipe Creator を導入した操作端末からタッチ操作により UI を操作し、レシピタイプや動作のきっかけとなる機器名や機器の状態、連携する機器名や動作内容などのレシピ作成に必要なレシピデータを選択する。入力されたレシピデータをもとに、レシピ作成 API ではレシピの動作条件となる要素と連携する機器の制御内容となる要素にまとめたレシピ情報が作成される。最後に、レシピ情報を Event, Condition, Action の要素ごとにまとめ、結合することでレシピが作成される。作成されたレシピはレシピ送信 API を介して iHAC Hub へ送信される。

3.2.2 iHAC Hub

レシピ受信およびレシピ実行の流れについて説明する。レシピ受信 API によってレシピを受け取った iHAC Hub は任意のパスに JSON 形式のファイルとしてレシピを書き込む。そのとき、同時にレシピ名の一覧が記述されたレシピリストに受信したレシピ名を書き込む。そして、レシピを解析するレシピエンジンがレシピリストを確認し、新たなレシピ名が追加されている場合、レシピを読み込み、機器制御を行う。

4 実装と評価

4.1 実装

iHAC Recipe Creator の UI の一部を図 3 に示す。iOS と Android の両プラットフォームに対応したアプリケーションを作成するため、JavaScript フレームワークである React をモバイル対応させた React Native を用いて実装を行なった。作成されたレシピを受けとるレシピ受信 API は Raspberry Pi3 に JavaScript を用いて実装した。iHAC Recipe Creator と iHAC Hub 間の通信は SSL 対応が容易である点や iHAC Hub 側から各部屋に設置されている機器の情報を iHAC Recipe Creator に送信できる拡張性の高さを考慮して Socket.io を用いて実装した。

4.2 評価

iHAC Recipe Creator のユーザビリティを評価するために、ISO9241-11 [3] によるユーザビリティの定義に基づいて評価を行なった。被験者は情報工学科の大学生（以下 IT 系とする）およびそれ以外の大学生（以下非 IT 系とする）各 10 名である。評価内容は「リビングのエアコンの電源が ON になったとき、リビングの照明の電源を ON にして光の色を昼白色にする」というレシピの作成を iHAC Recipe Creator を用いて実施した。このときにおけるレシピの正答率を算出し、正しくレシピが作

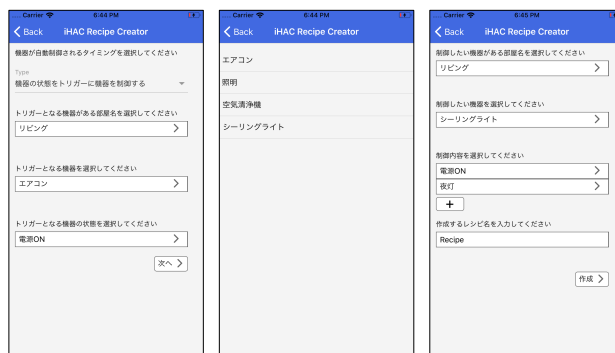


図 3 提案アプリケーションの UI

表 1 従来手法と提案手法の平均レシピ作成時間

	IT 系	非 IT 系
従来手法 [sec]	289.9	479.2
提案手法 [sec]	52.9	62.1

成できているか（有効性）について評価した。また、同様のレシピを従来手法であるキーボードを用いた手入力でのレシピ作成を被験者に対して実施した。この際、レシピを作成するまでの時間を提案手法と従来手法で比較し、時間的効率性が測れているかを評価した。

IT 系の正答率が 100%、非 IT 系は 90% であることから iHAC Recipe Creator を用いたレシピ作成方法は有効であると確認できた。表 1 に従来手法と提案手法の平均レシピ作成時間を示す。従来手法よりもレシピ作成時間を IT 系は 81.8%、非 IT 系は 87.0% 削減することができ、時間的効率性の観点から提案手法が優れていることを確認できた。

また、IT 系と非 IT 系を比較したとき、どちらの被験者に対しても iHAC Recipe Creator は有効性および時間的効率性があることから、提案手法は IT スキルに関係なく操作できることが確認できた。

5 まとめ

本稿では iHAC システムにおける機器連携レシピを作成する iHAC Recipe Creator の設計と実装を行なった。評価実験を行った結果、iHAC Recipe Creator を用いたレシピ作成方法はユーザビリティが高いことが確認できた。

参考文献

- [1] 梅山 他：情報処理学会論文誌コンシューマ・デバイスシステム Vol. 6, No. 1, pp. 84–93, 2016.
- [2] K. Hayashi and H. Suzuki, IEEE GCCE 2019, pp.806 – 807, Oct 2019.
- [3] ISO9241-11
<https://www.sis.se/api/document/preview/611299/>