

(19)日本国特許庁(JP)

(12)公開特許公報(A)

(11)公開番号
特開2025-67794
(P2025-67794A)

(43)公開日 令和7年4月24日(2025.4.24)

(51)国際特許分類
G 0 6 F 8/30 (2018.01)

F I
G 0 6 F 8/30

テーマコード (参考)
5 B 3 7 6

審査請求 未請求 請求項の数 14 O L (全36頁)

(21)出願番号 (22)出願日 (31)優先権主張番号 (32)優先日 (33)優先権主張国・地域又は機関 特願2024-128741(P2024-128741) 令和6年8月5日(2024.8.5) 特願2023-177815(P2023-177815) 令和5年10月13日(2023.10.13) 日本国(JP)	(71)出願人 (74)代理人 (72)発明者 Fターム (参考) 000006747 株式会社リコー 東京都大田区中馬込1丁目3番6号 110004381 弁理士法人I T O H 近藤 誠一 東京都大田区中馬込1丁目3番6号 株 式会社リコー内 5B376 BC31 BC46
---	--

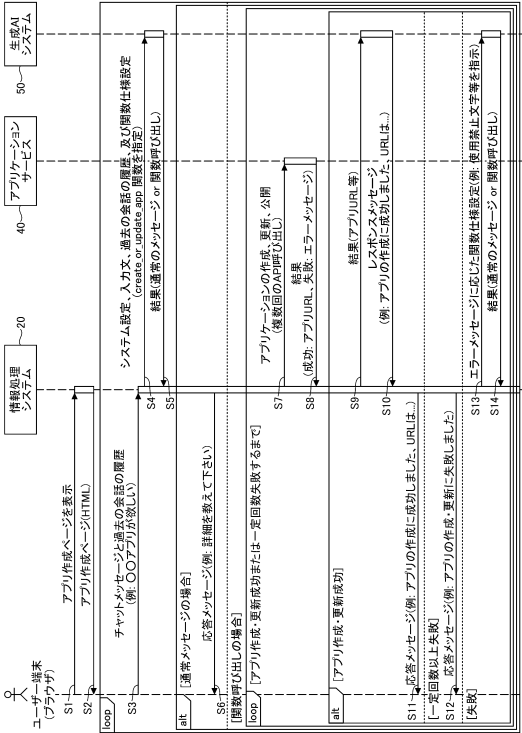
(54)【発明の名称】 情報処理装置、アプリ生成システム、プログラム、アプリケーション生成方法

(57)【要約】

【課題】 外部の生成A Iシステムを利用してアプリケーションを生成すること。

【解決手段】 本発明は、情報処理装置は、予め定義された関数の実行により機能するアプリケーション生成部であって、アプリケーションの生成に係る要求に含まれるパラメータに基づいて、該要求に係る前記アプリケーションを生成するアプリケーション生成部と、端末装置から送信されるチャットメッセージを受信した場合、受信した該チャットメッセージを、前記生成A Iシステムに送信するチャット制御部と、を有し、生成A Iシステムに送信した前記チャットメッセージと、設定情報と、関数を呼び出すための情報とに基づいて生成A Iシステムが送信した要求を、チャット制御部が受信すると、アプリケーション生成部が、受信した該要求に含まれるパラメータに基づいて、該要求に係る前記アプリケーションを生成する。

【選択図】 図10



【特許請求の範囲】

【請求項 1】

端末装置、及び、生成 A I (Artificial Intelligence) システムとネットワークを介して通信できる情報処理装置であって、

予め定義された関数の実行により機能するアプリケーション生成部であって、アプリケーションの生成に係る要求に含まれるパラメーターに基づいて、該要求に係る前記アプリケーションを生成するアプリケーション生成部と、

前記端末装置から送信されるチャットメッセージを受信した場合、受信した該チャットメッセージを、前記生成 A I システムに送信するチャット制御部と、

を有し、

前記チャット制御部は、少なくとも 1 回は、前記アプリケーションを生成するための設定情報と、前記関数を呼び出すための情報とを前記生成 A I システムに送信し、

前記生成 A I システムに送信した前記チャットメッセージと、前記生成 A I システムに送信した前記設定情報と、前記生成 A I システムに送信した前記関数を呼び出すための情報とに基づいて前記生成 A I システムが送信した前記要求を、前記チャット制御部が受信すると、

前記アプリケーション生成部が、受信した該要求に含まれる前記パラメーターに基づいて、該要求に係る前記アプリケーションを生成する、

情報処理装置。

【請求項 2】

前記チャット制御部は、前記生成 A I システムに送信した前記チャットメッセージに対する応答メッセージを前記生成 A I システムから受信した場合、該応答メッセージを前記端末装置に送信する、請求項 1 に記載の情報処理装置。

【請求項 3】

前記チャット制御部は、生成された前記アプリケーションにアクセスするためのアクセス情報を前記生成 A I システムが前記応答メッセージに含めることを可能とするために、該アクセス情報を前記生成 A I システムに送信する、請求項 2 に記載の情報処理装置。

【請求項 4】

前記チャット制御部は、前記アクセス情報を前記生成 A I システムに送信したことへの応答として、前記アクセス情報を含む前記応答メッセージを前記生成 A I システムから受信した場合、前記アクセス情報を含む前記応答メッセージを前記端末装置に送信する、請求項 3 に記載の情報処理装置。

【請求項 5】

前記関数を呼び出すための情報は、前記生成 A I システムが呼び出す関数の仕様の設定を含み、

関数の仕様の設定に基づいて前記生成 A I システムが前記情報処理装置に送信する前記関数を呼び出す関数呼び出し要求には、前記アプリケーションの生成に係る要求が含まれ、

前記チャット制御部は、前記アプリケーションの生成に係る要求を前記生成 A I システムから受信する、請求項 1 ～ 4 のいずれか 1 項に記載の情報処理装置。

【請求項 6】

前記チャット制御部が、前記アプリケーションの生成に係る要求を前記生成 A I システムから受信した場合、前記アプリケーション生成部は前記パラメーターに基づいて前記要求に係る前記アプリケーションを生成し、

前記チャット制御部が、前記アプリケーションの生成に係る要求でなく前記チャットメッセージに対する前記応答メッセージを前記生成 A I システムから受信した場合、

前記チャット制御部は前記応答メッセージを前記端末装置に送信する、請求項 3 に記載の情報処理装置。

【請求項 7】

前記アプリケーション生成部が前記アプリケーションの生成に失敗した場合、

前記チャット制御部は、前記アプリケーションの生成の失敗により取得したエラーメッセージ、及び、再度、受信した該チャットメッセージを、前記生成 A I システムに送信し、前記エラーメッセージ、及び、前記チャットメッセージに対する応答として前記アプリケーションの生成に係る要求を受信する、請求項 1 に記載の情報処理装置。

【請求項 8】

前記アプリケーション生成部が前記アプリケーションの生成に、一定回数、失敗した場合、

前記チャット制御部は、前記アプリケーションの生成の失敗した旨を前記端末装置に送信する、請求項 1 に記載の情報処理装置。

【請求項 9】

前記アプリケーションが表にデータが登録されるアプリケーションである場合、

前記パラメータは前記表の 1 つ以上のフィールド名である、請求項 1 に記載の情報処理装置。

【請求項 10】

前記情報処理装置はネットワークを介して前記アプリケーションを実行するアプリケーションサービスと通信し、

前記アプリケーション生成部は、前記パラメータを指定して、前記アプリケーションサービスに対し、前記アプリケーションの生成を要求する、請求項 1 に記載の情報処理装置。

【請求項 11】

前記端末装置では、前記アプリケーションサービスと通信する第一の Web ブラウザと、前記第一の Web ブラウザが起動させた第二の Web ブラウザとが実行され、

前記情報処理装置は、前記第二の Web ブラウザから前記チャットメッセージを受信し、

前記チャット制御部が、前記生成 A I システムが送信した前記アプリケーションの生成に係る要求を受信すると、

前記アプリケーション生成部は、前記第二の Web ブラウザに前記アプリケーションの生成に係る要求を送信し、

前記第二の Web ブラウザは前記第一の Web ブラウザに前記アプリケーションの生成に係る要求を送信し、

前記第一の Web ブラウザが前記アプリケーションサービスに対し、前記アプリケーションの生成に係る要求を送信する、請求項 10 に記載の情報処理装置。

【請求項 12】

端末装置、及び、生成 A I (Artificial Intelligence) システムと情報処理装置がネットワークを介して通信するアプリ生成システムであって、

前記端末装置は、

チャットメッセージを前記生成 A I システムに送信し、

前記情報処理装置は、

予め定義された関数の実行により機能するアプリケーション生成部であって、アプリケーションの生成に係る要求に含まれるパラメータに基づいて、該要求に係る前記アプリケーションを生成するアプリケーション生成部と、

前記端末装置から送信されるチャットメッセージを受信した場合、受信した該チャットメッセージを、前記生成 A I システムに送信するチャット制御部と、

を有し、

前記チャット制御部は、少なくとも 1 回は、前記アプリケーションを生成するための設定情報と、前記関数を呼び出すための情報とを前記生成 A I システムに送信し、

前記生成 A I システムに送信した前記チャットメッセージと、前記生成 A I システムに送信した前記設定情報と、前記生成 A I システムに送信した前記関数を呼び出すための情報とに基づいて前記生成 A I システムが送信した前記要求を、前記チャット制御部が受信すると、

10

20

30

40

50

前記アプリケーション生成部は、受信した該要求に含まれる前記パラメーターに基づいて、該要求に係る前記アプリケーションを生成する、
アプリ生成システム。

【請求項 13】

端末装置、及び、生成 A I (Artificial Intelligence) システムとネットワークを介して通信できる情報処理装置を、

予め定義された関数の実行により機能するアプリケーション生成部であって、アプリケーションの生成に係る要求に含まれるパラメーターに基づいて、該要求に係る前記アプリケーションを生成するアプリケーション生成部と、

前記端末装置から送信されるチャットメッセージを受信した場合、受信した該チャットメッセージを、前記生成 A I システムに送信するチャット制御部、
として機能させ、

前記チャット制御部は、少なくとも 1 回は、前記アプリケーションを生成するための設定情報と、前記関数を呼び出すための情報とを前記生成 A I システムに送信し、

前記生成 A I システムに送信した前記チャットメッセージと、前記生成 A I システムに送信した前記設定情報と、前記生成 A I システムに送信した前記関数を呼び出すための情報とに基づいて前記生成 A I システムが送信した前記要求を、前記チャット制御部が受信すると、

前記アプリケーション生成部が、受信した該要求に含まれる前記パラメーターに基づいて、該要求に係る前記アプリケーションを生成する、

プログラム。

【請求項 14】

端末装置、及び、生成 A I (Artificial Intelligence) システムとネットワークを介して通信できる情報処理装置が行うアプリケーション生成方法であって、

前記情報処理装置は、予め定義された関数の実行により機能するアプリケーション生成部であって、アプリケーションの生成に係る要求に含まれるパラメーターに基づいて、該要求に係る前記アプリケーションを生成するアプリケーション生成部と、

前記端末装置から送信されるチャットメッセージを受信した場合、受信した該チャットメッセージを、前記生成 A I システムに送信するチャット制御部と、
を有し、

前記チャット制御部は、少なくとも 1 回は、前記アプリケーションを生成するための設定情報と、前記関数を呼び出すための情報とを前記生成 A I システムに送信し、

前記生成 A I システムに送信した前記チャットメッセージと、前記生成 A I システムに送信した前記設定情報と、前記生成 A I システムに送信した前記関数を呼び出すための情報とに基づいて前記生成 A I システムが送信した前記要求を、前記チャット制御部が受信すると、

アプリケーション生成部が、受信した該要求に含まれる前記パラメーターに基づいて、該要求に係る前記アプリケーションを生成する、

アプリケーション生成方法。

【発明の詳細な説明】

【技術分野】

【0001】

本発明は、情報処理装置、アプリ生成システム、プログラム、及びアプリケーション生成方法に関する。

【背景技術】

【0002】

アプリケーションの開発には、開発者がソースコードをキーボード等から入力するプログラミング（コーディング）と呼ばれる作業が必要である。しかしながら、プログラム言語の文法や効率的なプログラミングを習熟するためには多くの時間やコストを必要とする。そこで、プログラムコードの入力を少なくしたり全くなくしたりするプログラミング支

10

20

30

40

50

援技術が知られている。

【0003】

人工知能（ＡＩ）を利用したプログラミング方法が考案されている（例えば、特許文献１参照）。特許文献１には、会話文からテキストデータを取得し、抽出した文脈に対して１以上のプログラムコードが関連付けて記憶されるプログラムコードデータベースと、プログラム文法解析データベースと、実現すべき処理動作に対して効率化されたプログラムが関連付けられたプログラム自動生成知識データベースとを参照し、抽出した文脈に応じた１以上のプログラムを生成する技術が開示されている。

【発明の概要】

【発明が解決しようとする課題】

10

【0004】

しかしながら、ＡＩシステムを利用してアプリケーション等のプログラムを生成するためには、当該アプリケーションとは別に、ＡＩシステムを開発、保守又は運用する必要がある。

【0005】

そこで、本発明は、上記課題に鑑み、外部のＡＩシステム（生成ＡＩシステム等）を利用してアプリケーションを生成することを目的とする。

【課題を解決するための手段】

【0006】

本発明は、上記課題に鑑み 端末装置、及び、生成ＡＩ（Artificial Intelligence）システムとネットワークを介して通信できる情報処理装置であって、予め定義された関数の実行により機能するアプリケーション生成部であって、アプリケーションの生成に係る要求に含まれるパラメーターに基づいて、該要求に係る前記アプリケーションを生成するアプリケーション生成部と、前記端末装置から送信されるチャットメッセージを受信した場合、受信した該チャットメッセージを、前記生成ＡＩシステムに送信するチャット制御部と、を有し、前記チャット制御部は、少なくとも１回は、前記アプリケーションを生成するための設定情報と、前記関数を呼び出すための情報とを前記生成ＡＩシステムに送信し、前記生成ＡＩシステムに送信した前記チャットメッセージと、前記生成ＡＩシステムに送信した前記設定情報と、前記生成ＡＩシステムに送信した前記関数を呼び出すための情報とに基づいて前記生成ＡＩシステムが送信した前記要求を、前記チャット制御部が受信すると、前記アプリケーション生成部が、受信した該要求に含まれる前記パラメーターに基づいて、該要求に係る前記アプリケーションを生成する。

20

30

【発明の効果】

【0007】

外部のＡＩシステム（生成ＡＩシステム等）を利用してアプリケーションを生成することができる。

【図面の簡単な説明】

【0008】

【図１】ユーザーが入力したチャットメッセージに対し生成ＡＩシステムが関数の実行を要求し、情報処理システムがアプリケーションを生成するまでの処理を説明する図である。

40

【図２Ａ】ユーザー端末が表示するアプリ作成ページの一例を示す図である。

【図２Ｂ】図２Ａの会話により作成された飲食店向けの在庫管理アプリが表示する飲食店向けの在庫管理アプリ画面の一例を示す図である。

【図３】アプリ生成システムの一例のシステム構成を示す図である。

【図４】情報処理システム、ユーザー端末、及び、開発者端末の一例のハードウェア構成を示す図である。

【図５】情報処理システム、ユーザー端末及び開発者端末の一例の機能構成を示す図である。

【図６】ユーザーがユーザー端末の画面上で「日報アプリを作ってください」と入力した場

50

合に、情報処理システムが生成 A I システムの A P I を呼び出すリクエストメッセージの一例を示す図である。

【図 7】図 6 のリクエストメッセージに対する生成 A I システムのレスポンスメッセージの一例を示す図である。

【図 8】情報処理システムがアプリケーションサービスに対しアプリケーションの生成又は更新を行った後に、生成 A I システムに送信するリクエストメッセージの一例を示す図である。

【図 9】図 8 のリクエストメッセージに対する生成 A I システムのレスポンスメッセージの一例を示す図である。

【図 10】情報処理システムがアプリケーション開発を支援する処理を説明するシーケンス図の一例である。

【図 11】ユーザーがアプリを開発する際に、ユーザー端末が表示するアプリ作成ページの表示例を説明する図である。

【図 12】図 11 の会話により作成された顧客管理アプリが表示する顧客管理アプリ画面の一例を示す図である。

【図 13】ユーザー端末が表示するアプリケーションサービスのポータル画面の一例を示す図である。

【図 14】ユーザー端末内の 2 つの W e b ブラウザが連携することを利用して、情報処理システムがアプリケーションサービスのアプリケーションを生成又は更新するシーケンス図の一例である（その 1）。

【図 15】ユーザー端末内の 2 つの W e b ブラウザが連携することを利用して、情報処理システムがアプリケーションサービスのアプリケーションを生成又は更新するシーケンス図の一例である（その 1）。

【図 16】ユーザーが入力したチャットメッセージに対し、生成 A I システムが関数の実行を情報処理システムに要求し、情報処理システムがアプリケーションを生成するまでの処理を説明する図である。

【図 17】アプリケーションサービスと情報処理システムとが一体の場合における、情報処理システム、ユーザー端末及び開発者端末の一例の機能構成を示す図である。

【発明を実施するための形態】

【0009】

以下、本発明を実施するための形態の一例として情報処理システムと、情報処理システムが行うアプリケーション生成方法について図面を参照しながら説明する。

【0010】

<生成 A I システムを用いたアプリケーション開発の支援>

ChatGPT（登録商標）等の生成 A I システムが知られている。生成 A I システムは、テキスト生成、質問応答、テキスト分類、感情分析、情報抽出、文章要約など、さまざまな自然言語処理タスクを実行できる。また、プログラムのコーディング中に次に書くべきコードの提案をする Copilot（登録商標）等の生成 A I システムも知られている。本実施形態では、ユーザーが生成 A I システムとの会話（チャットともいう）を通じてアプリケーションを生成する技術について説明する。なお、生成 A I システム又はその機能を単に A I（Artificial Intelligence）という場合がある。

【0011】

ここで、外部の生成 A I システムは、アプリケーションに係るサービスを提供するシステムとは異なるシステムとして提供されていればよく、アプリケーションに係るサービスと生成 A I システムとが、それぞれ同じ提供者により提供されていてもよい。

【0012】

まず、専門知識が乏しいユーザーがアプリを開発する場合、以下の 2 つが障壁になると言われている。

1. どんなアプリを作ったらいいかわからない。
2. 作りたいアプリがあってもどう作ればいいのかわからない。

10

20

30

40

50

【0013】

障壁1に関しては生成AIシステムがアイデアを出してくれるため容易に解決する。生成AIシステムはインターネット上の大量の文章を元に学習することが一般的であるため、様々な業種や業務の知識を持っている。そのため生成AIシステムはアプリの種類に関しても豊富なアイデアを提供できる。

【0014】

障壁2に関して、一般に、生成AIシステムを含めAIはアプリケーションを生成することはできない。そこで、本実施形態では、生成AIシステムが提供する、関数を呼び出す機能(function calling、function call等と呼ばれる機能)を利用し、ユーザーと生成AIシステムとの間に情報処理システムが介在することで障壁2を解決する。図1、図2を参照して説明する。

10

【0015】

図1は、ユーザーが入力したチャットメッセージに対し、生成AIシステム50が関数の実行を情報処理システム20に要求し、情報処理システム20がアプリケーションを生成するまでの処理を説明する図である。なお、情報処理システム20については後述するが、情報処理システム20はユーザーが生成AIシステム50を利用してアプリケーション開発する際のユーザーインターフェースを提供する。また、情報処理システム20は、ユーザー端末10から送信されるチャットメッセージを、生成AIシステム50及びアプリケーションサービス40が公開するAPI(Application Programming Interface)を呼び出すリクエストメッセージに変換して、生成AIシステム50及びアプリケーションサービス40に送信する。

20

【0016】

(1)ユーザー端末10はユーザーが入力したチャットメッセージを情報処理システム20に送信する。本実施形態の場合、チャットメッセージは例えば「日報アプリを作って」などである。なお、チャットメッセージは、「日報アプリを作って」というような直接的な指示だけでなく、「日報アプリが欲しい」等の願望を表すメッセージも含んでよい。

【0017】

(2)情報処理システム20は、ユーザーが入力したチャットメッセージを受信すると、生成AIシステムが提供するリクエストメッセージを受信するためのAPI経由で、受信したチャットメッセージをリクエストメッセージに含めて生成AIシステム50に送信する。ここで、情報処理システム20は、リクエストメッセージにfunction(関数)の仕様の指定も含めて送信する。function(関数)とは、指定された引数で決まった処理を行い、結果として戻り値を戻すプログラムインターフェースである。この関数は、アプリケーションサービス40で利用されるアプリケーションを生成又は更新することを生成AIシステム50が情報処理システム20に要求できるようにするための関数である。この関数の仕様の設定を情報処理システム20がリクエストメッセージに含めて生成AIシステム50に送信することにより、生成AIシステム50からの要求に応じて、情報処理システム20がローコード又はノーコードでアプリケーションを生成又は更新できるようになる。なお、このfunctionの呼び出しを生成AIシステム50が情報処理システム20に要求することを、function callと呼ぶ。また、補足的な情報として、情報処理システム20はアプリケーションがどのようなものであるかを生成AIシステム50に通知するシステム設定をリクエストメッセージに含めることができる。システム設定と関数の仕様の設定の詳細については後述するが、システム設定は生成したいアプリケーションに関する全般的なポリシーであるのに対し、関数の仕様の設定は生成AIシステム50が呼び出すfunctionの仕様を情報処理システム20が生成AIシステム50に対し指定する情報である。

30

40

【0018】

なお、情報処理システム20は、システム設定や関数の仕様の設定を、受信したチャットメッセージと同じリクエストメッセージに含めなくてもよい。例えば、情報処理システム20は、受信したチャットメッセージを含むリクエストメッセージとは別のリクエスト

50

メッセージに、システム設定や関数の仕様の設定を含めて生成 A I システム 5 0 に送信してもよい。また、情報処理システム 2 0 は、システム設定や、関数の仕様の設定も別々のリクエストメッセージに含めて生成 A I システム 5 0 に送信してもよい。

【0019】

なお、情報処理システム 2 0 は、システム設定や関数の仕様の設定を、受信したチャットメッセージをリクエストメッセージに含めて送信する都度送信してもよく、最初に一度だけ送信してもよい。

【0020】

(3) 生成 A I システム 5 0 は、送信されたリクエストメッセージに含まれるチャットメッセージによる会話の履歴、関数の仕様の設定及びシステム設定に基づいて、関数呼び出し要求 (function call) を含むレスポンスメッセージを情報処理システム 2 0 に送信する。ここで、生成 A I システム 5 0 は関数の呼び出しを情報処理システム 2 0 に対して要求するだけであり、実際に関数を実行するのは情報処理システム 2 0 であることに注意されたい。ただし、生成 A I システム 5 0 は、ユーザーが要求したアプリの種類に対応したフィールド名など (パラメーターの一例) をリクエストメッセージに含める。日報アプリのフィールド名等の例については図 2 にて説明する。

【0021】

なお、生成 A I システム 5 0 は、会話の履歴、関数の仕様の設定、システム設定に基づいて、関数呼び出し要求を含むレスポンスメッセージを情報処理システム 2 0 に必ず送信するとは限らない。例えば、生成 A I システム 5 0 は、質問を含むレスポンスメッセージや、チャットメッセージの内容を理解できなかった旨を含むレスポンスメッセージを情報処理システム 2 0 に送信することもある。

【0022】

(4) 情報処理システム 2 0 は、生成 A I システム 5 0 から送信されたレスポンスメッセージに含まれる関数呼び出し要求 (function call) に応じて、アプリケーションを生成する。どのようなアプリケーションを生成するかは (2) (3) で決まる。より詳細には、情報処理システム 2 0 は、当該関数呼び出し要求 (function call) の内容から、アプリケーションサービス 4 0 の A P I を呼び出すためのリクエストメッセージを生成し、生成したリクエストメッセージをアプリケーションサービス 4 0 に送信することである。本実施形態において、このようなリクエストメッセージの生成に係る処理を、「関数の実行」又は「関数を実行する」と表現する場合がある。本実施形態の場合、情報処理システム 2 0 は、関数呼び出し要求 (function call) に応じて、アプリケーションサービス 4 0 のアプリケーションを生成又は更新する。

【0023】

このように、ユーザーがユーザー端末 1 0 で、自然言語でチャットメッセージを入力するだけで、情報処理システム 2 0 が生成 A I システム 5 0 を用いてアプリケーションを開発できるため、障壁 2 も解消される。

【0024】

図 2 A のアプリ作成支援の画面は、ユーザー端末 1 0 が表示するアプリ作成ページの一例である。アプリ作成ページには、ユーザーが入力したチャットメッセージと、生成 A I システム 5 0 の応答メッセージとが交互に時系列に表示されている。なお、図 2 A は、チャットの履歴を全て画面上で表示している例を示しているが、会話の履歴が増えると、スクロールして表示する形でもよい。

【0025】

以下にチャットメッセージと、応答メッセージの詳細を示す。

【0026】

チャットメッセージ 3 0 1 は、ユーザーにより入力された「どんなアプリを作ることができますか」というチャットメッセージである。チャットメッセージ 3 0 1 に対する応答メッセージ 3 0 2 には、問い合わせアプリや経営者管理アプリなどいくつかのアプリが列挙されている。

10

20

30

40

50

【0027】

チャットメッセージ303は、ユーザーにより入力された「飲食店向けの在庫管理アプリを作ってみてください」というチャットメッセージである。チャットメッセージ303に対する応答メッセージ304は、アプリ作成に当たっての生成AIシステム50からの質問が列挙されている。応答メッセージ304は、具体的には、以下の質問を列挙する。

「アプリのフィールド定義を決めるために、以下の質問にお答え下さい。

在庫アイテムの名称は何ですか？

在庫アイテムの数量は数値で管理しますか？

在庫アイテムの単位は何ですか？

在庫アイテムのカテゴリはありますか？

在庫アイテムの賞味期限は管理しますか？

在庫アイテムの画像を添付する必要はありますか？」という質問が列挙される。

【0028】

チャットメッセージ305は、ユーザーにより入力された「適当に作ってみて下さい」というチャットメッセージである。応答メッセージ306は、チャットメッセージ305に対する応答メッセージであり、例えば、アプリを作成した結果が表示される。応答メッセージ306は、具体的には、以下の結果を表示する。

「飲食店向けの在庫管理アプリを作成しました。以下のURLからアクセスすることができます。

飲食店在庫管理アプリ（URLが埋め込まれている）306a

このアプリでは以下のフィールドが設定されています。

アイテム名：在庫アイテムの名称を入力するフィールドです。

数量：在庫アイテムの数量を数値で管理するフィールドです。

単位：在庫アイテムの単位を入力するフィールドです。

カテゴリ：在庫アイテムのカテゴリを選択するドロップダウンフィールドです。食材、飲料、調味料、その他のいずれかを選択できます。

賞味期限：在庫アイテムの賞味期限を入力する日付フィールドです。

画像：在庫アイテムの画像を添付するためのフィールドです。」である。

【0029】

ユーザーが、飲食店在庫管理アプリ306aを押下する操作を行うと、作成された飲食店在庫管理アプリの画面が表示される。このアプリの画面は、例えば、アイテム名、数量、単位、カテゴリ、賞味期限、及び、画像のフィールドを有する。

【0030】

図2Bは、図2Aの会話により作成された飲食店在庫管理アプリ（飲食店向けの在庫管理アプリ）が表示する飲食店向けの在庫管理アプリ画面290の一例である。飲食店向けの在庫管理アプリ画面290は、アイテム名292、数量293、カテゴリ294、賞味期限295、及び、画像296のフィールドを有している。これらは、ユーザーと生成AIシステム50との会話で生成されたフィールドである。したがって、ユーザーは生成AIシステム50と会話することで、フィールド名を一切挙げなくても、飲食店向けの在庫管理アプリを開発できる。

【0031】

ユーザーが保存ボタン291を押下すると、ユーザー端末10が飲食店向けの在庫管理アプリを保存する旨をアプリケーションサービス40に送信する。以降、ユーザーはポータル画面等から飲食店向けの在庫管理アプリを利用できる。

【0032】

このように、本実施形態のアプリ生成システム100は、ユーザーが自然言語で生成AIシステム50と会話することで、アプリケーションを開発できる。情報処理システム20が関数(function)を生成AIシステム50に対して指定するので、生成AIシステム50がアプリケーションのフィールド名等と共に関数呼び出しを情報処理システム20に要求できる。そして、情報処理システム20が実際にアプリケーションの生成又は更新

10

20

30

40

50

を行う。これによってユーザーから見れば「日報アプリを適当に作って」とメッセージを入力すれば、実際に日報アプリを開発するノーコード又はローコードプラットフォームを実現できる。

【0033】

<用語について>

アプリケーションとは、アプリケーションプログラムの略であり、コンピュータが任意の業務に応じて作成したプログラムである。OS (Operating System) がコンピュータの動作における基本的な機能やシステム (ファイルシステム、通信、表示制御等) を提供する汎用的なソフトウェアであるのに対し、アプリケーションはOS上で特定に機能を提供する。アプリケーションにはWebアプリとネイティブアプリがある。本実施形態では、どちらが開発されてもよい。

10

【0034】

APIはアプリケーション (ソフトウェア) のインターフェースなので、システム同士をつないで機能や仕組みを共有するための接点である。APIは、アプリケーション同士が互いに情報をやりとりするのに使用するインターフェースの仕様を定めている。コンピュータ同士のAPIは、HTTP (HTTPS) 通信によって一方のWebサイトが他方のWebサイトと通信する仕様である。通信により、一方のWebサイトが他方のWebサイトの機能を使用できる。コンピュータ同士のAPIをWeb-APIという場合がある。

【0035】

APIは特定の要求 (例えば、データの取得、更新、削除、加工など) を送信すると、応答に対する結果 (例えばデータ、更新結果、削除結果、加工結果など) を返す。要求をリクエストメッセージといい、結果をレスポンスメッセージという場合がある。APIを呼び出すとは、APIの仕様にしたがって要求を送信し、結果を取得することをいう。APIを呼び出すことを、実行する、操作する、叩く、又は使用する等という場合もある。

20

【0036】

関数を呼び出すための情報とは、情報処理システム20が提供する関数を、情報処理システムの外部、例えば生成AIシステム50から呼び出すことを可能とする情報である。関数を呼び出すための情報は、関数を呼び出すためのWeb-APIに関する情報を含む。

【0037】

関数を呼び出すためのWeb-APIに関する情報とは、生成AIシステム50がWeb-APIを呼び出す (利用する) ことができるようにするための情報である。本実施形態では、関数の仕様の設定 (以下、関数仕様設定という) という用語で説明される。関数仕様設定は生成AIシステム50が呼び出すfunction callの仕様を情報処理システム20が生成AIシステム50に対し指定する情報である。生成AIシステム50は関数の仕様の設定に基づいて関数を情報処理システム20に対し呼び出すので、情報処理システム20は関数を解析できる。

30

【0038】

関数とは、指定された引数で決まった処理を行い、結果として戻り値を戻すプログラムインターフェースである。関数は、その仕様や機能を開発者が設計できるのに対し、APIは生成AIシステム50やアプリケーションサービス40が利用してよいと公開した機能に関するAPIが予め用意されている。関数はAPIの仕様の1つであり、APIが関数の指定方法を定義している場合にAPIを呼び出すことで設定できる。また、関数の呼び出し又は関数の呼び出し要求とは、関数の実行を要求することをいう。

40

【0039】

チャットサービスとは、複数のユーザー同士がリアルタイムにメッセージを送信したり受信したりするサービスである。生成AIシステムは、ユーザーがAIとリアルタイムにメッセージを送信したり受信したりするサービスである。

【0040】

設定情報とは、アプリケーションの生成の全般的なポリシーであり、設定情報は、例え

50

ばアプリケーションの内容説明やヒント、又は、ユーザーに聞いてはいけない内容などである。設定情報は、生成AIシステム50に通知されることで所望のアプリケーションの生成が容易になる情報である。

【0041】

システム設定は、この設定情報の全体又は一部がAPIを呼び出すリクエストメッセージに変換された情報をいう。すなわち、システム設定は、設定情報の一部又は全部がAPIの書式に変換された情報である。

【0042】

アプリケーションの生成に係る要求に含まれるパラメーターとは、アプリケーションの挙動に影響を与える変数である。パラメーターの内容はアプリケーションによって様々であるが、例えば表形式やデータベースを扱うアプリケーションの場合、フィールド名やフィールドのデータ形式等がある。その他、行数、列数、行や列のサイズ、データ表示の際のフォントや色、フィールドの位置などがパラメーターであってもよい。

【0043】

ユーザーとは、アプリケーションサービス40が提供するアプリケーションを使用するエンドユーザーである。ユーザーはアプリケーションの開発を行うこともできる。開発者は、ノーコード又はローコードプログラミングによるアプリケーション開発のために必要な設定を情報処理システム20に対して行う者である。開発者は後述する設定情報を情報処理システム20に対して設定しておく。

【0044】

<システム構成例>

図3を参照して、本実施形態に係るアプリ生成システム100のシステム構成について説明する。図3は、アプリ生成システム100の一例のシステム構成を示す図である。図3に示すアプリ生成システム100は、情報処理システム20と、ユーザー端末10と、開発者端末60と、を有する。情報処理システム20とユーザー端末10、及び、情報処理システム20と開発者端末60は、ネットワークN1、N2を介して通信可能に接続されている。また、情報処理システム20は、ネットワークN1を介して生成AIシステム50及び外部システムであるアプリケーションサービス40と通信できる。更に、ユーザー端末10及び開発者端末60は、ネットワークN1、N2を介して生成AIシステム50及び外部システムであるアプリケーションサービス40と通信できる。

【0045】

ユーザー端末10と開発者端末60は、企業や自宅などの施設に配置されており、ネットワークN2に接続されている。ネットワークN2は、LAN、Wi-Fi（登録商標）、広域イーサネット（登録商標）、4G、5G、6G等の携帯電話網、などでよい。ネットワークN1は、インターネットやWAN等の広域的なネットワークである。ユーザー端末10と開発者端末60は常にネットワークN2に接続されている必要はなく、情報処理システム20、生成AIシステム50又はアプリケーションサービス40を使用する際に接続されればよい。

【0046】

生成AIシステム50は、ユーザーが会話形式で人工知能（AI）と会話するサービスを提供する。このような生成AIシステム50としては、大規模言語モデル（LLM：Large Language Models）を使用したシステムが知られている。大規模言語モデルは、大量のテキストデータを使ってトレーニングされた自然言語処理のモデルである。生成AIシステム50は、予め膨大なテキストを取り込み、そこからディープラーニングや強化学習等で知識を学習している。生成AIシステム50は、この知識を利用することで、チャットメッセージに対し応答メッセージを与える。生成AIシステム50は、ユーザーが入力したチャットメッセージ等を受信するインターフェースを有する。このインターフェースで受信するチャットメッセージ等をプロンプトと呼ぶ。プロンプトは、指示や質問などユーザーにより入力されたチャットメッセージを含む。また、プロンプトは、情報処理システム20により生成される指示や命令等を含んでもよい。チャットメッセージは、人

間が話す自然言語で構成された会話文や言葉などでよい。また、画像をチャットメッセージとすることもできる。チャットメッセージに対し文章を生成する生成ＡＩシステム５０を生成ＡＩという場合がある。本実施形態では、ユーザーが入力したチャットメッセージに対し、生成ＡＩシステム５０が回答する応答メッセージを利用して、情報処理システム２０がアプリケーションサービス４０において動作するアプリケーションを生成する。アプリケーションサービス４０において動作するアプリケーションは、アプリケーションサービス４０で動作するＷＥＢアプリと、ユーザー端末１０にインストールされるネイティブアプリとを含む。ユーザー端末１０にインストールされるネイティブアプリは、ユーザー端末１０で実行されると、アプリケーションサービス４０に接続し、アプリケーションサービス４０の機能を実行する。

10

【００４７】

生成ＡＩシステム５０は以下のような特徴を有している。

- １．生成ＡＩシステム５０は自然な流れで会話を進める。
 - ２．生成ＡＩシステム５０はユーザーが知見を持たない分野でもアイデアを膨らませて提案できる。
 - ３．生成ＡＩシステム５０は正確なプログラムコードを出力できる。
- したがって、情報処理システム２０は生成ＡＩシステム５０を介してユーザーとの会話を自然な流れで進めることができる。また、ユーザーは作りたいアプリを指示するだけで、そのアプリに必要なアイデアをもらうことができる。また、生成ＡＩシステム５０が呼び出す関数は正確（再現性が高い）なので、情報処理システム２０はユーザーが望むアプリケーションを生成しやすい。

20

【００４８】

生成ＡＩシステム５０としては、ＧＰＴ－３、ＧＰＴ－４、Transformer、BERT（いずれも登録商標）などのＬＬＭを使用したシステムが知られている。本実施形態のアプリ生成システム１００には一例としてＧＰＴ－３又はＧＰＴ－４を使用したChatGPTを使用可能であるが、その他のＬＬＭを使用したシステムが使用されてもよい。

【００４９】

アプリケーションサービス４０は、ユーザーが実行するアプリケーションを提供する一台以上の情報処理装置である。アプリケーションサービス４０が提供するアプリケーションは、例えばデータを表形式で管理するデータベース型のＷｅｂアプリである。ユーザーはアプリケーションのフィールドを任意に設定でき、自分の業務に関するデータを保存、読出、又は加工できるようにカスタマイズできる。このように予め用意されているデータベースにユーザーが所望するフィールド等を設定することがアプリケーションの開発である。本実施形態では、情報処理システム２０がユーザーからのチャットメッセージをリクエストメッセージに含めて生成ＡＩシステム５０に送信する。情報処理システム２０は、生成ＡＩシステム５０からのレスポンスメッセージを使用して、アプリケーションサービス４０に対しアプリケーションを生成又は更新する。すなわち、ユーザーはチャットメッセージを入力することで、ノーコードでアプリケーションサービス４０のアプリケーションを開発できる。

30

【００５０】

アプリケーションサービス４０と情報処理システム２０のドメインが異なる場合（例えばサービス元が異なる場合）、同じユーザーであってもアプリケーションサービス４０と情報処理システム２０ではアカウントが異なる。

40

【００５１】

一方、アプリケーションサービス４０と情報処理システム２０は一体でもよい。すなわち、アプリケーションサービス４０に情報処理システム２０の機能が含まれてもよいし、情報処理システム２０にアプリケーションサービス４０の機能が含まれていてもよい。また、アプリケーションサービス４０と生成ＡＩシステム５０が一体でもよい。また、生成ＡＩシステム５０と情報処理システム２０は一体でもよい。また、情報処理システム２０と、アプリケーションサービス４０と、生成ＡＩシステム５０とが一体でもよい。

50

【0052】

アプリケーションサービス40の一例には、例えば、クラウドサービス、ASP (Application Service Provider)、SaaS等があり、ネットワークを介して提供される各種のサービスが含まれてよい。例えば、サービスの一例としてデータベースの提供やストレージサービスがある。アプリケーションサービス40は、インターネット上に存在しても、オンプレミスに存在してもよい。

【0053】

情報処理システム20は、ユーザー端末10からのチャットメッセージを生成AIシステム50に伝える。情報処理システム20は、一台以上の情報処理装置に構成される。また、情報処理システム20は、ユーザー端末10から見て、要求を受け付けるサーバの役割である。サーバとは、クライアントからの要求に対して情報や処理結果を提供する機能を果たすコンピュータやソフトウェアである。情報処理システム20は、ユーザー端末10にはユーザーインターフェースを提供し、受信したチャットメッセージからリクエストメッセージを生成し、生成したリクエストメッセージを生成AIシステム50に送信する。また、情報処理システム20は生成AIシステム50からアプリケーション生成要求を受信すると、アプリケーションサービス40に対しアプリケーションの生成を要求する。ユーザーとしてはユーザー端末10を生成AIシステム50に接続させたり、アプリケーションサービス40に接続させたりする手間がない。

【0054】

なお、後述するように、ユーザーはアプリケーションサービス40をポータルにして、アプリケーション開発を行う場合がある。情報処理システム20は、アプリケーションサービス40と接続しているユーザー端末10から生成AIシステム50への接続要求を受け取り、生成AIシステム50と通信する。このような通信について詳細は図13～図15にて説明する。

【0055】

また、情報処理システム20は、開発者端末60から設定情報の設定を受け付ける。設定情報の設定とは、アプリケーションサービス40のアプリケーションの生成に必要な知識(設定情報)を情報処理システム20に設定することである。設定情報は、生成AIシステム50がアプリケーションのパラメータを生成するための情報である。設定情報は、例えばどのようなアプリケーションの生成が要求されるかを含んでよい。開発者が設定情報を情報処理システム20に設定しておくことで、ユーザーがチャットメッセージにアプリケーションのパラメータ等を含めなくてよい。情報処理システム20は生成するアプリケーションに関する適切なシステム設定を設定情報に基づいて作成し、作成したシステム設定を生成AIシステム50に通知しておくことができる。システム設定は設定情報から生成され、システム設定は設定情報の一部又は全部である。

【0056】

設定情報は、ユーザーがアプリケーションの開発を開始する前に情報処理システム20に設定されている。しかしながら、情報処理システム20が顧客ごとにカスタマイズすることも可能である。例えば、顧客の業種に応じて、設定情報が「あなたはこれから\$\$業のお客さんと会話します」である場合、情報処理システム20は、\$\$の業種名を顧客の業種に応じて変更して生成AIシステム50に通知する。

【0057】

情報処理システム20は、この設定情報を使用して、生成AIシステム50に送信するリクエストメッセージを作成する。リクエストメッセージにおいて、設定情報は後述するシステム設定に反映される。詳細は後述する。開発者が設定情報を情報処理システム20に送信しておくことで、ユーザーがアプリケーションの専門的な知識がなくても簡単なチャットメッセージでアプリケーションを生成できるようになる。

【0058】

なお、設定情報の設定は必ず必要なものではなく、アプリケーションによっては設定情報が不要の場合があってもよい。

10

20

30

40

50

【0059】

情報処理システム20は、クラウドコンピューティングにより実現されてもよいし、単一の情報処理装置によって実現されてもよい。クラウドコンピューティングとは、特定ハードウェア資源が意識されずにネットワーク上のリソースが利用される形態をいう。情報処理システム20は、インターネット上に存在しても、オンプレミスに存在してもよい。

【0060】

また、情報処理システム20は、複数の情報処理装置にその機能が分散されていてもよいし、複数の同じ機能を有する情報処理システム20が存在し、処理負荷に応じて、動画を配信する情報処理システムが増減されてもよい。

【0061】

ユーザー端末10又は開発者端末60は、例えば、ユーザー又は開発者が使用するPC(Personal Computer)、スマートフォン、タブレット端末等の端末装置である。ユーザー端末10又は開発者端末60では、Webブラウザ又はネイティブアプリが動作する。開発者は、開発者端末60を操作して、アプリケーションに関する設定情報を作成する。ユーザーは、ユーザー端末10を操作して、アプリケーションサービス40のアプリケーションを開発するためのチャットメッセージを作成する。管理者又はユーザーは開発者端末60及びユーザー端末10を操作して、情報処理システム20、生成AIシステム50又はアプリケーションサービス40が提供する各種のサービスを利用することができる。

【0062】

ユーザー端末10は、情報処理装置であればよい。このような情報処理装置としては、例えば、電子黒板、デジタルサイネージ等の出力装置、HUD(Head Up Display)装置、産業機械、撮像装置、集音装置、医療機器、ネットワーク家電、携帯電話、スマートフォン、タブレット端末、ゲーム機、PDA(Personal Digital Assistant)、デジタルカメラ、ウェアラブルPC等がある。

【0063】

<ハードウェア構成例>

図4を参照して、アプリ生成システム100に含まれる情報処理システム20、ユーザー端末10、及び、開発者端末60のハードウェア構成について説明する。生成AIシステム50とアプリケーションサービス40については、図4と同様か、又は、クラウドコンピューティングに対応した情報処理装置のハードウェア構成であるとする。

【0064】

<<情報処理システム、ユーザー端末、及び、開発者端末>>

図4は、本実施形態に係る情報処理システム20、ユーザー端末10、及び、開発者端末60の一例のハードウェア構成を示す図である。図4に示されているように、情報処理システム20、ユーザー端末10、及び、開発者端末60はコンピュータ500によって構築されており、CPU501、ROM502、RAM503、HD(Hard Disk)504、HDD(Hard Disk Drive)コントローラ505、ディスプレイ506、外部機器接続I/F(Interface)508、ネットワークI/F509、バスライン510、キーボード511、ポインティングデバイス512、DVD-RW(Digital Versatile Disk Rewritable)ドライブ514、メディアI/F516を備えている。

【0065】

これらのうち、CPU501は、コンピュータ500全体の動作を制御する。ROM502は、IPL等のCPU501の駆動に用いられるプログラムを記憶する。RAM503は、CPU501のワークエリアとして使用される。HD504は、プログラム等の各種データを記憶する。HDDコントローラ505は、CPU501の制御にしたがってHD504に対する各種データの読み出し又は書き込みを制御する。ディスプレイ506は、カーソル、メニュー、ウィンドウ、文字、又は画像などの各種情報を表示する。外部機器接続I/F508は、各種の外部機器を接続するためのインターフェースである。この場合の外部機器は、例えば、USB(Universal Serial Bus)メモリやプリンタ等であ

10

20

30

40

50

る。ネットワーク I / F 5 0 9 は、ネットワーク N 2 を利用してデータ通信をするためのインターフェースである。バスライン 5 1 0 は、図 4 に示されている C P U 5 0 1 等の各構成要素を電氣的に接続するためのアドレスバスやデータバス等である。

【 0 0 6 6 】

また、キーボード 5 1 1 は、文字、数値、又は各種指示などの入力に使用される複数のキーを備えた入力手段の一種である。ポインティングデバイス 5 1 2 は、各種指示の選択や実行、処理対象の選択、カーソルの移動などを行う入力手段の一種である。光学ドライブ 5 1 4 は、着脱可能な記録媒体の一例としての D V D - R W 5 1 3 に対する各種データの読み出し又は書き込みを制御する。なお、着脱可能な記録媒体は、D V D - R W 5 1 3 に限らず、D V D - R 等であってもよい。メディア I / F 5 1 6 は、フラッシュメモリ等の記録メディア 5 1 5 に対するデータの読み出し又は書き込み（記憶）を制御する。

10

【 0 0 6 7 】

< 機能について >

次に、図 5 を参照し、アプリ生成システム 1 0 0 の機能構成について説明する。図 5 は、情報処理システム 2 0 、ユーザー端末 1 0 及び開発者端末 6 0 の一例の機能構成を示す図である。

【 0 0 6 8 】

<< ユーザー端末 >>

ユーザー端末 1 0 は、通信部 1 1 と、表示制御部 1 2 と、操作受付部 1 3 と、ブラウザ間連携部 1 4 とを有する。これら各機能部は、ユーザー端末 1 0 にインストールされた 1 以上のプログラムに含まれる命令を図 4 に示した C P U 5 0 1 が実行することで実現される機能又は手段である。例えば、通信部 1 1 、表示制御部 1 2 、操作受付部 1 3 及びブラウザ間連携部 1 4 は W e b ブラウザ及び JavaScript（登録商標。以下省略する。）の少なくとも一方により実現されてよい。開発者端末 6 0 がネイティブアプリを実行する場合、これらはネイティブアプリにより実現されてよい。

20

【 0 0 6 9 】

通信部 1 1 は、情報処理システム 2 0 との間で各種の情報を送受信する。本実施形態では、通信部 1 1 は、ポータル画面の画面情報をアプリケーションサービス 4 0 から受信したり、チャットメッセージの入力が行われるアプリ作成ページ等の画面情報を情報処理システム 2 0 から受信したりする。通信部 1 1 は、チャットメッセージやユーザー端末 1 0 に対する操作情報等を情報処理システム 2 0 に送信する。なお、通信部 1 1 が生成 A I システム 5 0 と直接通信することもできる。

30

【 0 0 7 0 】

表示制御部 1 2 は、各種の画面の画面情報を解釈してディスプレイ 5 0 6 に表示する。操作受付部 1 3 は、ディスプレイ 5 0 6 に表示された各種画面におけるユーザーの各種操作を受け付ける。

【 0 0 7 1 】

ブラウザ間連携部 1 4 は、W e b ブラウザ間で情報の送受信を行う。1 つのユーザー端末 1 0 は同時に複数の W e b ブラウザを実行することが可能である。ブラウザ間連携部 1 4 は、複数のうち一方の W e b ブラウザから他方の W e b ブラウザに（及びこの逆に）情報を通知できる。ブラウザ間連携部 1 4 により、一方の W e b ブラウザのみが特定の W e b サイト（例えばアプリケーションサービス 4 0）について認証情報を保持している場合、他方の W e b ブラウザは一方の W e b ブラウザと連携して、特定の W e b サイトと通信できる。一方の W e b ブラウザはアプリケーションサービス 4 0 と通信し、他方の W e b ブラウザは情報処理システム 2 0 を介して生成 A I システム 5 0 と通信する W e b ブラウザであるとする。他方の W e b ブラウザは生成 A I システム 5 0 からのレスポンスメッセージを一方の W e b ブラウザに送信できる。ブラウザ間連携部 1 4 の機能は、W e b ブラウザ（ユーザー端末 1 0）が実行するものであるが、アプリケーションサービス 4 0 及び情報処理システム 2 0 から送信される画面情報にブラウザ間連携部 1 4 を実現する JavaScript で記述されている。

40

50

【0072】

<<開発者端末>>

開発者端末60は、通信部61と、表示制御部62と、操作受付部63と、を有する。これら各機能部は、開発者端末60にインストールされた1以上のプログラムに含まれる命令を図4に示したCPU501が実行することで実現される機能又は手段である。例えば、通信部61、表示制御部62及び操作受付部63はWebブラウザ又はJavaScriptの少なくとも一方により実現されてよい。開発者端末60がネイティブアプリを実行する場合、これらはネイティブアプリにより実現されてよい。

【0073】

通信部61は、情報処理システム20との間で各種の情報を送受信する。本実施形態では、通信部61は、設定情報の入力に関する画面情報を情報処理システム20から受信する。通信部61は、設定情報や開発者端末60に対する操作情報等を情報処理システム20に送信する。なお、通信部21が生成AIシステム50やアプリケーションサービス40と通信することもできる。

10

【0074】

表示制御部62は、各種の画面の画面情報を解釈してディスプレイ506に表示する。操作受付部63は、ディスプレイ506に表示された各種画面におけるユーザーの各種操作を受け付ける。

【0075】

<<情報処理システム>>

情報処理システム20は、通信部21、画面生成部22、チャット制御部23、設定情報記憶部24、及びアプリケーション生成部25、を有している。情報処理システム20が有する各機能部は、情報処理システム20にインストールされた1以上のプログラムに含まれる命令を図4に示したCPU501が実行することで実現される機能又は手段である。また、設定情報記憶部24は、図4に示したHD504やRAM503等に構築される。設定情報記憶部24は情報処理システム20が有していなくてもよく、情報処理システム20がアクセスできるネットワーク上にあればよい。

20

【0076】

通信部11は、ユーザー端末10と各種の情報を送受信する。通信部11は、アプリ作成ページの画面情報をユーザー端末10に送信する。通信部11は、ユーザー端末10からユーザーがアプリケーション開発のために入力したチャットメッセージを受信する。また、通信部11は、開発者端末60と各種の情報を送受信する。通信部11は、設定情報の入力画面の画面情報を開発者端末60に送信する。通信部11は、開発者端末60から開発者が入力した設定情報を受信し設定情報記憶部24に保存しておく。

30

【0077】

画面生成部22は、ユーザー端末10及び開発者端末60が表示する画面(Webページ)の画面情報を生成する。WebページはWebアプリの一部として提供される。画面情報は、HTML、JSON、XML、スクリプト言語、及びCSS(Cascading Style Sheet)等で記述されたプログラムであり、主にHTMLによりWebページの構造が特定され、スクリプト言語によりWebページの動作が規定され、CSSによりWebページのスタイルが特定される。ユーザー端末10がネイティブアプリを実行する場合もある。ネイティブアプリとは、ユーザー端末10にインストールしなければ実行できないアプリケーションである。ネイティブアプリの場合、画面情報はユーザー端末10が保持しており、表示される情報がJSONやXML等で送信される。

40

【0078】

チャット制御部23は、第1API26を呼び出すことで、生成AIシステム50にリクエストメッセージを送信する。リクエストメッセージには、チャットメッセージ(過去の会話の履歴も含む)や関数仕様設定が含まれる。第1API26は、生成AIシステム50が公開する1つ以上のAPIである。第1API26を呼び出した結果として、チャット制御部23は、リクエストメッセージに対するレスポンスメッセージを生成AIシス

50

テム 50 から受信する。レスポンスメッセージにはチャットメッセージに対する応答メッセージ、情報処理システム 20 に対する関数呼び出し要求、又は、作成若しくは開発したアプリケーションの URL 等が含まれる。

【0079】

アプリケーション生成部 25 は、第 2 API 27 を呼び出すことで、アプリケーションサービス 40 にリクエストメッセージを送信する。アプリケーション生成部 25 は、予め定義された関数の実行により機能する。リクエストメッセージには、アプリケーションに関するパラメーター（データベースのフィールド名、データ形式等）などが含まれる。アプリケーションサービス 40 に対するリクエストメッセージは、アプリケーションの生成又は更新の要求を行うものである。第 2 API 27 は、アプリケーションサービス 40 が公開する 1 つ以上の API である。第 2 API 27 を呼び出した結果として、アプリケーション生成部 25 は、リクエストメッセージに対するレスポンスメッセージをアプリケーションサービス 40 から受信する。レスポンスメッセージにはアプリケーションの生成又は更新の要求に対する結果（成功、失敗）が含まれる。成功の場合はアプリケーションの URL が通知され、失敗の場合はエラーメッセージが通知される。

10

【0080】

なお、アプリケーション生成部 25 は、アプリケーションサービス 40 に要求せずに、情報処理システム 20 内でアプリケーションを生成してもよい。アプリケーション生成部 25 は、例えば、表計算ソフト上で動作するマクロを生成してもよいし、データベースソフトにフィールドを設定してもよい。

20

【0081】

ここで、「アプリケーションを生成する」とは、アプリケーション生成部 25 が実際にアプリケーションを生成することだけでなく、アプリケーション生成部 25 が、アプリケーションサービス 40 等にアプリケーションの生成を依頼して、アプリケーションサービス 40 等が実際にアプリケーションを生成することも含む。

【0082】

＜生成 AI システムの動作に関する補足＞

本実施形態では、情報処理システム 20 は、生成 AI システム 50 が公開する第 1 API 26 を呼び出すことで生成 AI システム 50 を利用する。情報処理システム 20 が第 1 API 26 を呼び出すことで、生成 AI システム 50 の機能の 1 つである AI との会話が可能になる。生成 AI システム 50 には特徴的な動作又は処理があるためそれらについて補足する。

30

【0083】

情報処理システム 20 は、API を呼び出すためのリクエストメッセージを生成 AI システム 50 に送信することで生成 AI システム 50 と通信する。リクエストメッセージにより、生成 AI システム 50 からのレスポンスメッセージを情報処理システム 20 が取得できる。

【0084】

生成 AI システム 50 はリクエストメッセージに対するレスポンスメッセージを情報処理システム 20 に返す。生成 AI システム 50 のレスポンスメッセージに含まれる応答メッセージは、同じチャットメッセージであっても都度異なることが多い。ただし、情報処理システム 20 が温度と呼ばれるパラメーターを 0 に指定すれば同じチャットメッセージに対して、生成 AI システム 50 が同じ応答メッセージを返す。

40

【0085】

情報処理システム 20 が生成 AI システム 50 との会話を続けたい場合、過去の会話の履歴を全てリクエストメッセージに含める必要がある。生成 AI システム 50 との会話はステートレスだからである。これは、ある生成 AI システム 50 の仕様であり、別の生成 AI システム 50 では過去の会話の履歴を含めなくてよい場合もある。

【0086】

このように、第 1 API 26 は過去の通信に関する状態を持たない(ステートレス)。こ

50

のため、過去の会話の履歴は第1API26の利用者（本実施形態では、情報処理システム20）が保持し、第1API26を呼び出す際に毎回指定する必要がある。過去の会話の履歴はユーザー端末10が保持してもよい。また、会話履歴が長くなりすぎると第1API26はエラーを返すため、情報処理システム20は会話が長く続いた場合は直近のやりとりのみをリクエストメッセージに含めるなど、適宜、工夫することが好ましい。

【0087】

<<生成AIシステムが返すfunction call>>

生成AIシステム50が提供する機能には、function callと呼ばれる機能がある。function callとは、生成AIシステム50が情報処理システム20に対し関数（function）の呼び出し（call）を要求することをいう。よって、情報処理システム20がこの機能

10

【0088】

生成AIシステム50が「関数」を呼び出すように、リクエストメッセージには情報処理システム20が設定する関数仕様設定が含まれる。関数仕様設定により生成AIシステム50がfunction call（関数呼び出し要求）することが可能になる。このように関数仕様設定は、アプリケーション生成部25を呼び出すための第1API26に関する情報である。すなわち、関数仕様設定は、関数の呼び出しを生成AIシステム50が情報処理システム20に要求できることを、情報処理システム20が生成AIシステム50に通知する。

20

【0089】

このように、情報処理システム20が生成AIシステム50にfunction callを使用させることで、情報処理システム20が提供する関数を生成AIシステム50が呼び出し、情報処理システム20は関数を実行できる。また、情報処理システム20は、関数の実行結果を利用して、ユーザーと会話することが可能になる。

【0090】

ただし、生成AIシステム50のレスポンスメッセージは毎回異なるので、生成AIシステム50が関数呼び出し要求するとは限らない。情報処理システム20がリクエストメッセージに関数仕様設定を含めても、関数仕様設定がない場合と同じレスポンスメッセージを生成AIシステム50が返す場合もある。

30

【0091】

過去の会話の履歴を全てリクエストメッセージに含める処理を採用すると、情報処理システム20が関数を実行した実行結果もリクエストメッセージに含めて生成AIシステム50に送信する必要がある。ただし、情報処理システム20が、過去の会話の履歴と共に、関数の実行結果をリクエストメッセージに含めて生成AIシステム50に送信することは必須でない。すなわち、情報処理システム20が関数の実行結果をそのまま、ユーザー端末10に送信してもよい。しかし、情報処理システム20が関数の実行結果を生成AIシステム50に送信することで、生成AIシステム50が過去の会話の履歴を使用して会話を継続できるので、会話に連続性が生まれる。例えば文体や言葉遣いに一貫性が生じるので、ユーザーが違和感を覚えにくくなる。例えば、ユーザーの母国語が英語で、英語で会話していたとしても、情報処理システム20が応答メッセージを英語に翻訳する必要もない。

40

【0092】

また、情報処理システム20は生成AIシステム50が関数呼び出しを要求するかどうか予測できない。しかし、情報処理システム20がリクエストメッセージの中で明示的に「function_callオプション」を指定すれば、生成AIシステム50に対し、関数呼び出しすることを強制できる。

【0093】

<アプリケーションサービスのアプリケーション開発の支援>

本実施形態では、ユーザーが日報アプリを開発する場合を例にして、情報処理システム

50

20が送信するリクエストメッセージや生成AIシステム50が送信するレスポンスメッセージについて説明する。

【0094】

図6は、ユーザーがユーザー端末10の画面上で「日報アプリを作ってください」と入力した場合に、情報処理システム20が、生成AIシステム50が公開する第1API26を呼び出すリクエストメッセージの一例である。リクエストメッセージにはrole230aとcontent230bが含まれている。role230aは発言者を示し、user(ユーザー)、assistant(生成AIシステム50のAI)、及び、system(AIアシスタントの設定を指示)がある。content230bは会話の内容であり、自然言語で記述されてよい。

【0095】

図6のリクエストメッセージは、まず、systemによるシステム設定229を含む。システム設定229は、「あなたには今から〇〇アプリケーションサービスと呼ばれるSaaSサービス内のアプリ開発を支援してもらいます。あなたにはこれから利用者と対話してアプリケーションの「フィールド」を決めてもらいます。フィールドの定義はJSON形式で表現します。例えば以下のような3つのフィールドを持つ日報アプリが考えられます」という情報229aである。また、システム設定229は、フィールド名として「日付」「業務内容」「所感、学び」というフィールドを指示する情報229bを含む。各フィールドはそれぞれDATE、MULTI_LINE_TEXT、MULTI_LINE_TEXTのようにデータ形式も指示されている。これらの情報229a、229bは、開発者が事前に情報処理システム20に送信した設定情報に基づいている。

【0096】

また、リクエストメッセージは、ユーザーが入力した「日報アプリを作ってください」というチャットメッセージ231を含む。

【0097】

また、図6のリクエストメッセージは関数仕様設定233を含む。functionsが関数仕様設定233であり、関数仕様設定233は、以下の属性を有する。

name234は、関数の名称(create_or_update_app)である。

description235は、関数の仕様を指示する。仕様の詳細は、「日報アプリを作成します。成功時は{"ok": true, "value": "アプリのURL"}を返し、失敗時は{"ok": false, "error": ...}を返します。」である。すなわち、description235はURL又はエラーメッセージを返すことを生成AIシステム50に指示している。

parameters236も関数の仕様の一部である。

"type": "object"は戻り値のデータ形式を示す。

"required": ["properties", "name"]は戻り値に要求される項目名を示す。

"name": { "type": "string" }はnameのデータ形式を示す。

"type": "array"は表形式であることを示す。

"items"以下は、フィールドに必要な属性とデータ形式を示す。

【0098】

この関数仕様設定233により、図6のように定義された関数の呼び出しを生成AIシステム50が情報処理システム20に対して要求できることを、情報処理システム20が生成AIシステム50に指示した。

【0099】

図7は、図6のリクエストメッセージに対する生成AIシステム50のレスポンスメッセージの一例である。レスポンスメッセージに関数呼び出し要求237(function call)が含まれている。本実施形態では、function callがレスポンスメッセージに含まれると、情報処理システム20がアプリケーションサービス40のアプリケーションを生成する。アプリケーションサービス40はローコード又はノーコードプラットフォームである。アプリケーションサービス40のアプリケーションは、ユーザーのデータを管理するデータベースであり、データを表示させるUI画面をユーザーが容易にカスタマイズでき

10

20

30

40

50

るものである。アプリケーションサービス40は様々なAPIを公開しており、情報処理システム20は、第2API27を呼び出すことでアプリケーションの生成や更新を行うことが可能である。本実施形態では、アプリケーションサービス40が公開する第2API27と生成AIシステム50が公開する第2API27を呼び出して、ユーザーが生成AIシステム50と自然言語で会話する中で、情報処理システム20がアプリケーションサービス40のアプリケーションを生成又は更新することを可能にする。

【0100】

レスポンスメッセージの関数呼び出し要求237はアプリケーション生成要求に相当する。関数呼び出し要求237には、name238とarguments239が含まれる。name238は関数仕様設定233で定義された関数の名称である。生成AIシステム50はこの名称の関数の呼び出しを要求している。arguments239は戻り値であり、関数仕様設定233で定義されている。図7では、「日報」というアプリ名239a、日報アプリが含むフィールド名239b、239cとこれらのデータ形式(DATE、MULTI_LINE_TEXT)等が戻り値239に含まれている。

【0101】

生成AIシステム50はこのfunction callにより、業務内容などのフィールド名を指定し、アプリケーションを生成する関数の呼び出しを情報処理システム20に要求できる。

【0102】

情報処理システム20は、レスポンスメッセージに含まれるfunction callを解析して、アプリケーションサービス40のアプリケーションを生成・更新すると判断する(関数の実行)。このため、情報処理システム20は、アプリケーションサービス40が公開する第2API27を呼び出し、レスポンスメッセージに含まれるフィールド名やデータ形式(パラメーターの一例)を使って、アプリケーションの生成又は更新を行う。アプリケーションの生成又は更新に成功した場合、情報処理システム20は、アプリケーションサービス40からアプリケーションのURLを取得する。URLは、ユーザー端末10がアプリケーションにアクセスするためのアクセス情報である。情報処理システム20は、生成AIシステム50にアプリケーションのURLを返す。こうすることで、ユーザーと生成AIシステム50との会話に連続性が生まれる。

【0103】

具体的には以下のように情報処理システム20がアプリケーションサービス40の3つのAPIを使用する例が考えられる。

1. アプリケーションサービス40が公開するアプリケーションを生成するAPI
2. 1で作成したアプリケーションに"日付"と"業務内容"という2つのフィールドを追加するAPI
3. アプリケーションを公開するAPI

アプリケーションの生成又は更新に成功した場合、情報処理システム20は、関数の結果(アプリケーションのURL)を指定して再度生成AIシステム50が公開する第1API26を呼び出す。

【0104】

図8は、情報処理システム20がアプリケーションサービス40に対しアプリケーションの生成又は更新を行った後に、生成AIシステム50に送信するリクエストメッセージの一例である。このリクエストメッセージは、過去の会話の履歴として、システム設定229、チャットメッセージ231、及び、関数呼び出し要求241が含まれる。また、情報処理システム20は、関数実行結果243を過去の会話の履歴に追加する。図8の関数実行結果243は、関数を実行したこと243a、実行した関数の名称243b(create_or_update_app)及びURL243cである。なお、図8には再度、関数仕様設定233が含まれるが、これも過去の会話の履歴である。図8には、過去の会話の履歴として関数仕様設定233を含めているが、生成AIシステム50が提供するAPIの仕様によっては含めなくてもよい。

10

20

30

40

50

【0105】

なお、図8では、URL243c以外に、過去の会話の履歴（システム設定229、チャットメッセージ231、関数呼び出し要求241、関数を実行したこと243a、関数の名称243b）が含まれているが、過去の会話の履歴を生成AIシステム50が保持する機能がある場合、情報処理システム20はURL243cのみを生成AIシステム50に送信すればよい。すなわち、過去の会話の履歴の内容は過去に少なくとも1回、情報処理システム20が生成AIシステム50に送信しておけばよい場合もあり得る。

【0106】

図9は、図8のリクエストメッセージに対する生成AIシステム50のレスポンスメッセージの一例である。なお、アプリケーション開発の場合もレスポンスメッセージは毎回異なる場合が多い。例えば、図9のようなレスポンスメッセージが返されることが期待される。図9のレスポンスメッセージには「アプリが作成されました。\"https://sample.com/k/42/\" から確認して下さい」という応答メッセージ245が含まれる。情報処理システム20は、このURLをユーザー端末10に送信するので、ユーザーはURLを押下することで、開発したアプリケーション（データベース）のフィールド名などを確認できる。

10

【0107】

以降は、ユーザーが満足するまで、チャットメッセージの入力と、情報処理システム20と生成AIシステム50の間でリクエストメッセージとレスポンスメッセージが送受信されることが繰り返される。このように、ユーザーが会話形式でアプリケーションサービス40のアプリケーションを開発できるため、ユーザーはプログラミングの知識は一切必要ない。

20

【0108】

＜動作又は処理＞

図10は、情報処理システム20がアプリケーション開発を支援する処理を説明するシーケンス図の一例である。大きな流れとしてステップS3～S14が繰り返し実行され、その中でステップS9～S14が繰り返し実行される。

【0109】

S1：ユーザー端末10の通信部11が情報処理システム20のURLに接続する。このURLは、ユーザー端末10で、ユーザーが直接入力する方法や、URLが指定されたショートカットを実行することで指定してもよい。ユーザー端末10の通信部11はアプリ作成ページのURLを含むHTTPリクエストを生成し、生成したHTTPリクエストを情報処理システム20に送信する。

30

【0110】

S2：情報処理システム20の通信部21が、ユーザー端末10から上記のHTTPリクエストを受信すると、画面生成部22がアプリ作成ページの画面情報を作成する。通信部21は、上記のHTTPリクエストの応答として、ユーザー端末10にアプリ作成ページの画面情報を含むHTTPレスポンスを送信する。図11にアプリ作成ページの一例を示す。

【0111】

S3：ユーザーはユーザー端末10にチャットメッセージ（例えば〇〇アプリを作成して下さい）を入力する。入力はキーボードを使用してもよいし、音声入力でもよい。操作受付部13はチャットメッセージを受け付ける。ユーザー端末10の通信部11はチャットメッセージを含めたリクエストメッセージを生成し、生成したリクエストメッセージを情報処理システム20に送信する。ユーザー端末10の通信部11は、生成したリクエストメッセージを、例えば、HTTPリクエストとして送信する。なお、ユーザー端末10の通信部11は過去の会話の履歴を含めたリクエストメッセージを生成してよい。また、過去の会話の履歴は情報処理システム20が保持しておいてもよい。過去の会話の履歴を情報処理システム20が保持している場合は、ユーザー端末10の通信部11は、リクエストメッセージに過去の会話の履歴を含めなくてもよい。

40

50

【0112】

S4：情報処理システム20の通信部21はチャットメッセージと過去の会話の履歴を受信する。情報処理システム20の通信部21はチャットメッセージと過去の会話の履歴をチャット制御部23に渡す。チャット制御部23は、システム設定（設定情報）、チャットメッセージ、過去の会話の履歴、及び関数仕様設定を含むリクエストメッセージを生成し、生成したリクエストメッセージを、第1API26を呼び出すことにより生成AIシステム50に送信する（図6参照）。

【0113】

S5：生成AIシステム50が応答メッセージ又は関数呼び出し要求を含むレスポンスメッセージを情報処理システム20に返す（図7参照）。チャット制御部23はレスポンスメッセージを生成AIシステム50から受信する。このレスポンスメッセージは応答メッセージの場合と関数呼び出し要求の場合がある。関数呼び出し要求の場合、処理はステップS7に進み、応答メッセージの場合、処理はステップS6に進む。

10

【0114】

S6：チャット制御部23が通常の応答メッセージを受信した場合、情報処理システム20の通信部21は、応答メッセージをユーザー端末10に送信する。例えば応答メッセージが「詳細を教えてください」の場合である。

【0115】

S7：チャット制御部23が関数呼び出し要求を受信した場合、この関数を解析し、アプリケーションサービス40のアプリケーションの生成又は更新が要求（アプリケーション生成要求）されたと判断する。ステップS4の関数仕様設定233でアプリケーション生成のための関数が指定されているので、関数呼び出し要求に含まれる関数の名称（name238）に基づいて、チャット制御部23はアプリケーションの生成要求であることを容易に判断できる。また、チャット制御部23は関数呼び出し要求に含まれるarguments239（アプリ名239a、フィールド名239b、239c、データ形式等）を取得する。

20

【0116】

このため、チャット制御部23は、アプリケーションサービス40のアプリケーションの生成又は更新をアプリケーション生成部25に要求する。アプリケーション生成部25は第2API27を呼び出すことで、アプリの作成又は更新を行うリクエストメッセージをアプリケーションサービス40に対して送信する。すなわち、アプリケーション生成部25は、リクエストメッセージを送信することで生成AIシステム50から呼び出し要求された関数の機能を実現する。本実施形態では、アプリケーション生成部25はアプリケーションを生成又は更新する。アプリケーション生成部25は上記の3つのAPIを呼び出す。

30

1. アプリケーションサービス40が公開するアプリケーションを生成するAPI
2. 1で作成したアプリケーションに"日付"と"業務内容"などの複数のフィールドを追加するAPI
3. アプリケーションを公開するAPI

"2."において、アプリケーション生成部25は、リクエストメッセージに、生成AIシステム50からのレスポンスメッセージに含まれるパラメーターである、arguments239（アプリ名239a、フィールド名239b、239c、データ形式等）を含め、アプリケーションの生成を要求する。具体的なリクエストメッセージについてはアプリケーションサービス40によって異なってよい。

40

【0117】

なお、情報処理システム20は、2回目の関数呼び出し要求の際は既存アプリの更新を行う。このためアプリの作成を行うだけでなく更新も可能となる。

【0118】

S8：アプリケーション生成部25はレスポンスメッセージをアプリケーションサービス40から受信する。このレスポンスメッセージが含む結果は、例えば「アプリの作成に

50

成功しました。URLは……」などである。また、アプリケーションの生成に失敗した場合、アプリケーションサービス40からエラーメッセージが返される。成功の場合、処理はステップS9に進み、失敗の場合、処理はステップS13に進む。

【0119】

S9：アプリ生成に成功した場合、チャット制御部23は、第1API26を呼び出すことで、アプリケーションのURLを含むリクエストメッセージを生成AIシステム50に送信する（図8参照）。こうすることでユーザーと生成AIシステム50との会話に連続性が生じる。

【0120】

S10：生成AIシステム50が応答メッセージを含むレスポンスメッセージを情報処理システム20に返す。チャット制御部23はレスポンスメッセージを生成AIシステム50から受信する（図9参照）。このレスポンスメッセージにはアプリケーションのURLが含まれる。

【0121】

S11：アプリ生成に成功した場合、情報処理システム20の通信部21は、ステップS10のレスポンスメッセージに含まれる応答メッセージをユーザー端末10に送信する。図10の処理はいったん終了し、この後、ユーザーはアプリケーションを更新するため、必要に応じてステップS3に戻る。

【0122】

S12：アプリ生成に失敗し、そして、一定回数以上失敗した場合、情報処理システム20の通信部21はユーザー端末10に「アプリの作成・更新に失敗しました」等の結果を送信する。図10の処理はいったん終了し、この後、ユーザーはアプリケーション再度、作成するため、必要に応じてステップS3に戻る。

【0123】

S13：ステップS8のレスポンスメッセージがアプリ生成の失敗を通知するものである場合、チャット制御部23は、第1API26を呼び出すことで、アプリケーションサービス40からのエラーメッセージを生成AIシステム50に送信する。エラーメッセージに加え過去の会話の履歴も送信される。こうすることで、生成AIシステム50はエラーメッセージが出ない関数呼び出し要求を作成できる。例えば、アプリケーション生成部25がアプリケーションサービス40に送信したリクエストメッセージが、アプリケーションサービス40が公開する第2API27の仕様に違反している場合、エラーメッセージに正しい仕様の内容が含まれる場合がある。生成AIシステム50が仕様を理解してエラーメッセージが出ない関数呼び出し要求に近づけることができる。例えば、エラーメッセージがアプリケーションサービス40で使用できない文字種の指摘である場合、エラーメッセージに使用できない文字種が含まれるので、生成AIシステム50が該文字種を含まない関数呼び出し要求を作成できるようになる。文字種に限らず、リクエストメッセージの書式についても同様である。

【0124】

S14：チャット制御部23はレスポンスメッセージを生成AIシステム50から受信する。レスポンスメッセージは関数呼び出し要求又は応答メッセージである。しかし、チャット制御部23がエラーメッセージを送信したので、アプリケーションの生成に成功する関数の呼び出し要求がレスポンスメッセージに含まれることが期待される。関数呼び出し要求の場合、処理はステップS7に進み、応答メッセージの場合、処理はステップS6に進む。

【0125】

<ユーザー端末のアプリ作成ページの表示例>

図11は、ユーザーがアプリを開発する際に、ユーザー端末10が表示するアプリ作成ページ310の表示例を説明する図である。アプリ作成ページ310は、ユーザーが入力したチャットメッセージと、生成AIシステム50の応答メッセージが交互に表示されている。チャットメッセージ311は「顧客管理アプリを適当に作ってみて下さい」である。

チャットメッセージ 3 1 1 に対する応答メッセージ 3 1 2 は、「顧客管理アプリを作成しました。以下の URL からアクセスすることができます。顧客管理アプリ（URL が埋め込まれている）

作成したアプリには以下のフィールドが含まれています。

顧客名、メールアドレス、電話番号、住所、会社名、業種、ステータス、メモ」

チャットメッセージ 3 1 3 は、「他にどんな入力項目を追加することが考えられますか？」である。チャットメッセージ 3 1 3 に対する応答メッセージ 3 1 4 は、「顧客管理アプリに追加できる入力項目はさまざまです。以下にいくつかの例を挙げます。

誕生日：顧客の誕生日を管理するためのフィールドです。

担当者：顧客を担当する社員を選択するためのフィールドです。

売上予測：顧客の売上予測を管理するためのフィールドです。

最終連絡日：顧客との最終連絡日を管理するためのフィールドです。

重要度：顧客の重要度を管理するためのフィールドです。」である。

【0 1 2 6】

チャットメッセージ 3 1 5 は、「担当者と重要度を追加して？」である。チャットメッセージ 3 1 5 に対する応答メッセージ 3 1 6 は、「顧客管理アプリに「担当者」と「重要度」のフィールドを追加しました。以下の URL からアクセスすることができます。

顧客管理アプリ（URL が埋め込まれている）

アプリにアクセスしてフィールドの追加やデータの入力を行ってください。追加の要望や質問があればお知らせ下さい」

応答メッセージ 3 1 6 の「顧客管理アプリ」というアプリ名 3 1 7 に図 1 0 のステップ 8 でアプリケーションサービス 4 0 から返された URL が含まれている。ユーザーは URL を押下することで作成又は更新されたアプリケーションを使用できる。

【0 1 2 7】

チャットメッセージ 3 1 1, 3 1 3, 3 1 5 を見れば分かるように、ユーザーは顧客管理アプリのフィールド名を入力していない。チャットメッセージ 3 1 5 のフィールド名も生成 AI システム 5 0 が提案したものである。したがって、ユーザーはプログラミングの知識だけでなく顧客管理アプリの知識さえなくても顧客管理アプリを開発できた。

【0 1 2 8】

図 1 2 は、図 1 1 の会話により作成された顧客管理アプリが表示する顧客管理アプリ画面 2 6 0 の一例である。顧客管理アプリ画面 2 6 0 は、顧客名 2 6 1、メールアドレス 2 6 2、電話番号 2 6 3、住所 2 6 4、会社名 2 6 5、業種 2 6 6、ステータス 2 6 7、メモ 2 6 8、担当者 2 6 9、及び重要度 2 7 0 のフィールドを有している。これらは、ユーザーと生成 AI システム 5 0 との会話で生成されたフィールドである。したがって、ユーザーは生成 AI システム 5 0 と会話することで、フィールド名を一切挙げなくても、顧客管理アプリを開発できた。

【0 1 2 9】

なお、上記の例では、フィールド名を一切挙げない形で顧客管理アプリを生成する例を説明したが、ユーザーがチャットメッセージで、1 以上のフィールド名を指定する形であってもよい。

【0 1 3 0】

ユーザーが保存ボタン 2 7 1 を押下すると、ユーザー端末 1 0 が顧客管理アプリを保存する旨をアプリケーションサービス 4 0 に送信する。以降、ユーザーはポータル画面等から顧客管理アプリを利用できる。

【0 1 3 1】

<アプリケーションサービスが公開する第 2 API を情報処理システムが使用するための認証に関する補足>

情報処理システム 2 0 が第 2 API 2 7 を呼び出すためには情報処理システム 2 0 がアプリケーションサービス 4 0 から認証を受けることが必要になる。図 1 0 のシーケンス図の情報処理システム 2 0 がアプリケーションサービス 4 0 と同じドメインで動作している場

10

20

30

40

50

合、ユーザー端末10で動作するWebブラウザから情報処理システム20にアプリケーションサービス40の認証情報が送信される。認証情報は例えば予めcookieに含まれている。このため、アプリケーションサービス40が公開する第2API27を情報処理システム20が呼び出すことが可能である。ローコード又はノーコードプラットフォーム（アプリケーションサービス40）と情報処理システム20が同一の組織によって開発される場合は同じドメインにすることは容易であると考えられ、この場合、課題はないと考えられる。

【0132】

しかし、情報処理システム20とアプリケーションサービス40が別のドメインで動作する場合には認証に関する課題が生じうる。図13を参照して説明する。図13はユーザー端末10が表示するアプリケーションサービス40のポータル画面330の一例である。ユーザーはポータル画面330からアプリケーションを使用したり、アプリケーションを開発したりできる。

10

【0133】

例えば、アプリケーションサービス40のポータル画面330の構成が以下のようであると考えられる。ポータル画面330には「AIアプリ作成」ボタン331が表示されている。ユーザーが「AIアプリ作成」ボタン331を押下すると、ポップアップで情報処理システム20と通信する別のWebページ332が開かれるものとする。この場合、アプリケーションサービス40と情報処理システム20が別のドメインなので、ユーザー端末10は情報処理システム20にアプリケーションサービス40の認証情報を送信できない。したがって、情報処理システム20はアプリケーションサービス40が公開する第2API27を呼び出すことはできない。これは、ポップアップ表示された別のWebページ332は、ポータル画面330とはドメインが異なるので、ポップアップ表示された別のWebページ332はアプリケーションサービス40のcookieにアクセスできないためである。

20

【0134】

そこで、ポップアップ表示された別のWebページ332から元のポータル画面330に対し、アプリケーションサービス40が公開する第2API27の呼び出しを依頼する形を取る構成が考えられる。

【0135】

図14、図15は、ユーザー端末10内の2つのWebブラウザが連携することを利用して、情報処理システム20がアプリケーションサービス40のアプリケーションを生成又は更新するシーケンス図の一例である。なお、図14、図15の説明では主に図10との相違を説明する場合がある。大きな流れとしてステップS36～S49が繰り返し実行される。

30

【0136】

S21：ユーザーがユーザー端末10に対しアプリケーションサービス40のURLを指示する。アプリケーションサービス40に接続するWebブラウザを元のWebブラウザ240（第一のWebブラウザの一例）とする。

【0137】

S22：ユーザー端末10の通信部11がアプリケーションサービス40のURLに接続する。ユーザーはポータル画面330のURLをユーザー端末10に指示するので、ユーザー端末10のWebブラウザはポータル画面330のURLを含むHTTPリクエストをアプリケーションサービス40に送信する。

40

【0138】

S23：アプリケーションサービス40は、ユーザー端末10から上記のHTTPリクエストを受信すると、ポータル画面330の画面情報（HTML、JavaScript、CSS等）をユーザー端末10に送信する。この画面情報には、別のWebブラウザ250（第二のWebブラウザの一例）を起動させるスクリプト、及び、別のWebページ332（アプリ作成ページ）のURL等が含まれている。

50

【0139】

S24：ユーザー端末10の通信部11はポータル画面330の画面情報を受信し、元のWebブラウザ240の表示制御部12がポータル画面330を表示させる。

【0140】

S25：ユーザーはポータル画面330の「AIアプリ作成」ボタン331を押下する。

【0141】

S26：これによりブラウザ間連携部14は別のWebブラウザ250を起動させる。別のWebブラウザ250がポップアップ表示される。より詳細には、別のWebブラウザ250の起動は、ポータル画面330の画面情報に含まれるJavaScriptが行う。

【0142】

S27：別のWebブラウザ250は元のWebブラウザ240から通知された、別のWebページ332（アプリ作成ページ）のURLを使用して、別のWebページ332（アプリ作成ページ）を情報処理システム20に要求する。

【0143】

S28：情報処理システム20の画面生成部22は、アプリ作成ページの画面情報（HTML、JavaScript、CSS等）を生成し、通信部21はアプリ作成ページの画面情報を別のWebブラウザ250（ユーザー端末10）に送信する。

【0144】

S29：ユーザー端末10の通信部11はアプリ作成ページの画面情報を受信する。表示制御部12が画面情報に基づいてアプリ作成ページを表示させる。

【0145】

S30：ユーザーが例えば「○○アプリが欲しい」などのチャットメッセージをユーザー端末10に入力する。

【0146】

ユーザー端末10がアプリ作成ページを表示した後の流れは図10と同様になるので、図14、図15では、主に図10との相違を説明する。まず、ステップS31～S34は図10のステップS3～S6と同様でよい。

【0147】

S35：ステップS34で別のWebブラウザ250が、生成AIシステム50からのレスポンスメッセージ（ここでは応答メッセージの場合）を受信すると、別のWebブラウザ250の表示制御部12が応答メッセージ（例えば詳細を教えてください）を表示する。

【0148】

S36：ステップS34で別のWebブラウザ250が、生成AIシステム50からのレスポンスメッセージ（ここでは関数呼び出し要求の場合）を受信すると、別のWebブラウザ250のブラウザ間連携部14はアプリケーションサービス40のアプリケーションの生成又は更新を元のWebブラウザ240に要求する。このようなブラウザ間の連携（メッセージ通信）としては

<https://developer.mozilla.org/ja/docs/web-API/Window/postMessage>が知られている。

【0149】

S37：元のWebブラウザ240は、アプリケーションサービス40のアプリケーションの生成又は更新をアプリケーションサービス40に対して行う。

【0150】

S38：アプリケーションサービス40はアプリケーションの生成又は更新に関する結果（成功、失敗）を元のWebブラウザ240に送信する。成功の場合はアプリケーションのURLが通知される。失敗の場合はエラーメッセージが通知される。

【0151】

S39：元のWebブラウザ240のブラウザ間連携部14はアプリケーションの生成又は更新に関する結果を別のWebブラウザ250に送信する。この後、結果が成功の場合、処理はステップS40に進み、結果が失敗の場合、処理はステップS46に進む。

10

20

30

40

50

【0152】

S40：別のWebブラウザ250のブラウザ間連携部14は成功（アプリのURL）の結果を受信し、別のWebブラウザ250の通信部11はアプリケーションの生成又は更新に関する結果（成功）を情報処理システム20に送信する。したがって、図15の流れでは、情報処理システム20がアプリケーションサービス40ではなくユーザー端末10（別のWebブラウザ250）から結果を受信する。以降のステップS41～S43の処理は図10のステップS9～S11と同様である。

【0153】

S44：別のWebブラウザ250の通信部11は応答メッセージを受信する。表示制御部12が応答メッセージをアプリ作成ページに表示させる。ステップS44で図15の処理はいったん終了し、この後、ユーザーはアプリケーションを更新するため、必要に応じてステップS30に戻る。

【0154】

S45：ステップS39の結果（成功、失敗）を別のWebブラウザ250が受信した場合、ブラウザ間連携部14は、一定回数以上連続して失敗したかを判断する。一定回数以上連続して失敗した場合、表示制御部12が「アプリの作成・更新に失敗しました」等の結果を別のWebブラウザ250に表示させる。図15の処理はいったん終了し、この後、ユーザーはアプリケーションを再度、作成するため、必要に応じてステップS30に戻る。

【0155】

S46：別のWebブラウザ250のブラウザ間連携部14は失敗（エラーメッセージ）の結果を受信した場合、別のWebブラウザ250の通信部11はアプリケーションの生成又は更新に関する結果（失敗）を情報処理システム20に送信する。以降のステップS47、S48の処理は図10のステップS13,14と同様である。

【0156】

S49：情報処理システム20は別のWebブラウザ250に対し関数呼び出し要求又は通常の応答メッセージを送信する。関数呼び出し要求の場合、処理はステップS36に進み、応答メッセージの場合、処理はステップS35に進む。

【0157】

以降、繰り返し処理の際も情報処理システム20はユーザー端末10に依頼する形でアプリケーションサービス40のアプリケーションの生成又は更新を行う。

【0158】

また、図14、図15では、別のWebブラウザ250をポップアップ表示する例を説明したが、Webブラウザが対応するiframeと呼ばれる機能を利用してもよい。iframe（インラインフレーム）とは、HTMLの要素（タグ）で定義されており、Webページ内に別のWebページやコンテンツ（テキスト、画像、動画など）を表示させる機能である。Webページ内に別のWebページを埋め込むことで、アプリケーションサービス40のポータル画面内に別サイトであるアプリ作成ページを表示させることができる。この場合のウィンドウ間の通信はpostMessageが知られている。なお、この場合のシーケンス図は図10と同様でよい。

【0159】

<主な効果>

本実施形態の情報処理システム20は、専門知識が乏しいユーザーがアプリを開発する場合の、2つが障壁（どんなアプリを作ったらいいかわからない、作りたいアプリがあってもどう作ればいいかわからない）を解決できる。

【0160】

本実施形態のアプリ生成システム100は、ユーザーが自然言語で生成AIシステム50と会話することで、アプリケーションを開発できる。情報処理システム20が関数（function）を生成AIシステム50に対して指定するので、生成AIシステム50がアプリケーションのフィールド名等と共に関数呼び出しを情報処理システム20に要求できる

10

20

30

40

50

。そして、情報処理システム 20 が実際にアプリケーションの生成又は更新を行う。これによってユーザーから見れば「日報アプリを適当に作って」とメッセージを入力すれば、実際に日報アプリを開発するノーコード又はローコードプラットフォームを実現できる。

【0161】

＜その他の適用例＞

以上、本発明を実施するための最良の形態について実施例を用いて説明したが、本発明はこうした実施例に何等限定されるものではなく、本発明の要旨を逸脱しない範囲内において種々の変形及び置換を加えることができる。

【0162】

例えば、本実施形態では、ユーザーが入力するチャットメッセージが自然言語の例を説明したが、ユーザーは例えばアプリの画面を表す画像（写真、手書き、スナップショット等）を情報処理システム 20 に送信して、アプリケーションの生成を要求してもよい。情報処理システム 20 はアプリの画面を生成 AI システム 50 に送信し、生成 AI システム 50 はアプリの画面を解析して関数の呼び出し要求を情報処理システム 20 に送信する。生成 AI システム 50 はアプリの画面から抽出したフィールド名やデータ形式を関数の呼び出し要求に含めることができる。

10

【0163】

例えば、本実施形態では、ユーザーが入力するチャットメッセージが自然言語の例を説明したが、発話による音声で入力する方式でもよい。音声で入力する方式の場合、ユーザー端末 10 で発話により入力された音声を、ユーザー端末 10 から情報処理システム 20 に送信する。そして、情報処理システム 20 は、受信した音声をテキストに変換し、変換したテキストを含めたリクエストメッセージを生成する。そして、生成したリクエストメッセージを、生成 AI システム 50 に送信する。

20

【0164】

また、生成 AI システム 50 が音声の入力に対応している場合、情報処理システム 20 は、受信した音声を生成 AI システム 50 に送信してもよい。

【0165】

また、本実施形態では、ユーザーが例えば「○○アプリが欲しい」というアプリケーションの生成を明示的に情報処理システム 20 に要求しているが、ユーザーが必ずしも明示的に要求をしなくてもよい。例えば、ユーザーが「データ入力が面倒」というチャットメッセージを情報処理システム 20 に送信した場合に、情報処理システム 20 又は生成 AI システム 50 が困りごとを検出する。情報処理システム 20 又は生成 AI システム 50 が困りごとをアプリケーションの生成で解決できると判断すると、新しくアプリケーションを生成することをユーザーに提案できる。

30

【0166】

また、本実施形態では、情報処理システム 20 がアプリケーションの URL を生成 AI システム 50 に送信しているが、URL はアプリケーションにアクセスするためのアクセス情報であればよい。例えば、アプリケーションサービス 40 が生成したアプリケーションの ID（識別情報）を情報処理システム 20 に返した場合、情報処理システム 20 はアプリケーションの ID を生成 AI システム 50 に送信する。情報処理システム 20 は生成 AI システム 50 から返されたレスポンスメッセージに含まれるアプリケーションの ID をユーザー端末 10 に送信する。ユーザー端末 10 がアプリケーションの ID を指示してアプリケーションの実行を要求した場合、情報処理システム 20 が ID に基づいてアプリケーションの URL に変換し、この URL をユーザー端末 10 に返す。

40

【0167】

また、本実施形態では、情報処理システム 20 が生成 AI システム 50 から関数の呼び出し要求を受信する場合を例にして説明したが、情報処理システム 20 は Web-API そのものの呼び出し要求を生成 AI システム 50 から受信してもよい。すなわち、関数と API とは厳密に区別されなくてもよい。

【0168】

50

また、実施例に記載された装置群は、本明細書に開示された実施形態を実施するための複数のコンピューティング環境のうちの1つを示すものにすぎない。ある実施形態では、情報処理システム20は、サーバクラスタといった複数のコンピューティングデバイスを含む。複数のコンピューティングデバイスは、ネットワークや共有メモリなどを含む任意のタイプの通信リンクを介して互いに通信するように構成されており、本明細書に開示された処理を実施する。

【0169】

更に、情報処理システム20は、本実施形態で開示された処理ステップ、例えば図10等を様々な組み合わせで共有するように構成できる。例えば、所定のユニットによって実行されるプロセスは、情報処理システム20が有する複数の情報処理装置によって実行され得る。また、情報処理システム20は、1つのサーバ装置にまとめられていてもよいし、複数の装置に分けられていてもよい。

10

【0170】

また、図5などの構成例は、情報処理システム20による処理の理解を容易にするために、主な機能に応じて分割したものである。処理単位の分割の仕方や名称によって本願発明が制限されることはない。情報処理システム20の処理は、処理内容に応じて更に多くの処理単位に分割することもできる。また、1つの処理単位が更に多くの処理を含むように分割することもできる。

【0171】

また、上記で説明した実施形態の各機能は、一又は複数の処理回路によって実現することが可能である。ここで、本明細書における「処理回路」とは、電子回路により実装されるプロセッサのようにソフトウェアによって各機能を実行するようプログラミングされたプロセッサや、上記で説明した各機能を実行するよう設計されたASIC(Application Specific Integrated Circuit)、DSP(digital signal processor)、FPGA(field programmable gate array)や従来の回路モジュール等のデバイスを含むものとする。

20

【0172】

アプリケーションサービス40と情報処理システム20とが一体の場合の構成例を説明する。以下では、情報処理システム20にアプリケーションサービス40の機能が含まれる例を説明するが、アプリケーションサービス40に情報処理システム20の機能が含まれてもよい。

30

【0173】

図16は、ユーザーが入力したチャットメッセージに対し、生成AIシステム50が複数の実行を情報処理システム20に要求し、情報処理システム20がアプリケーションを生成するまでの処理を説明する図である。(1)~(3)の処理は図1の説明と同様でよい。

【0174】

(4) 情報処理システム20は、生成AIシステム50から送信されたレスポンスメッセージに含まれる関数呼び出し要求(function call)に応じて、アプリケーションを生成する。どのようなアプリケーションを生成するかは(2)(3)で決まる。より詳細には、情報処理システム20は、当該関数呼び出し要求(function call)の内容から、アプリケーションを生成するための関数を実行する。すなわち、図16の構成の場合、情報処理システム20は、自身がアプリケーションを生成又は更新する。

40

【0175】

図17は、アプリケーションサービス40と情報処理システム20とが一体の場合における、情報処理システム20、ユーザー端末10及び開発者端末60の一例の機能構成を示す図である。なお、図17の説明では主に図5との相違を説明する。情報処理システム20とアプリケーションサービス40が一体なので、アプリケーション生成部25は第2API27を呼び出さない。また、通信部21、画面生成部22、設定情報記憶部24、及び、チャット制御部23の機能は図5と同様でよい。

【0176】

そして図17のアプリケーション生成部25は、予め定義された関数の実行により機能

50

する。実行する関数に入力するパラメーターとしては、データベースのフィールド名、データ形式等などが含まれる。

【0177】

本発明の実施形態は、コンピュータの能力及び機能性に大きな改善をもたらす。これらの改善により、ユーザーは、情報処理装置において情報を格納し提示する方法であるテーブルとの、より効率的で堅牢な対話を提供するコンピュータを利用することができる。更に、本発明の実施形態は、より効率的で強力かつ堅牢なユーザーインターフェースの使用を通じて、より良いユーザー体験を提供する。このようなユーザーインターフェースは、人間と機械との間のより良い相互作用を提供する。

【0178】

<付記>

[付記1]

端末装置、及び、生成AI（Artificial Intelligence）システムとネットワークを介して通信できる情報処理装置であって、

予め定義された関数の実行により機能するアプリケーション生成部であって、アプリケーションの生成に係る要求に含まれるパラメーターに基づいて、該要求に係る前記アプリケーションを生成するアプリケーション生成部と、

前記端末装置から送信されるチャットメッセージを受信した場合、受信した該チャットメッセージを、前記生成AIシステムに送信するチャット制御部と、

を有し、

前記チャット制御部は、少なくとも1回は、前記アプリケーションを生成するための設定情報と、前記関数を呼び出すための情報とを前記生成AIシステムに送信し、

前記生成AIシステムに送信した前記チャットメッセージと、前記生成AIシステムに送信した前記設定情報と、前記生成AIシステムに送信した前記関数を呼び出すための情報とに基づいて前記生成AIシステムが送信した前記要求を、前記チャット制御部が受信すると、

前記アプリケーション生成部が、受信した該要求に含まれる前記パラメーターに基づいて、該要求に係る前記アプリケーションを生成する、

情報処理装置。

[付記2]

前記チャット制御部は、前記生成AIシステムに送信した前記チャットメッセージに対する応答メッセージを前記生成AIシステムから受信した場合、該応答メッセージを前記端末装置に送信する、付記1に記載の情報処理装置。

[付記3]

前記チャット制御部は、生成された前記アプリケーションにアクセスするためのアクセス情報を前記生成AIシステムが前記応答メッセージに含めることを可能とするために、該アクセス情報を前記生成AIシステムに送信する、付記2に記載の情報処理装置。

[付記4]

前記チャット制御部は、前記アクセス情報を前記生成AIシステムに送信したことへの応答として、前記アクセス情報を含む前記応答メッセージを前記生成AIシステムから受信した場合、前記アクセス情報を含む前記応答メッセージを前記端末装置に送信する、付記3に記載の情報処理装置。

[付記5]

前記関数を呼び出すための情報は、前記生成AIシステムが呼び出す関数の仕様の設定を含み、

関数の仕様の設定に基づいて前記生成AIシステムが前記情報処理装置に送信する前記関数を呼び出す関数呼び出し要求には、前記アプリケーションの生成に係る要求が含まれ、

前記チャット制御部は、前記アプリケーションの生成に係る要求を前記生成AIシステムから受信する、付記1～4のいずれか1項に記載の情報処理装置。

10

20

30

40

50

[付記 6]

前記チャット制御部が、前記アプリケーションの生成に係る要求を前記生成 A I システムから受信した場合、前記アプリケーション生成部は前記パラメータに基づいて前記要求に係る前記アプリケーションを生成し、

前記チャット制御部が、前記アプリケーションの生成に係る要求でなく前記チャットメッセージに対する前記応答メッセージを前記生成 A I システムから受信した場合、

前記チャット制御部は前記応答メッセージを前記端末装置に送信する、付記 3 に記載の情報処理装置。

[付記 7]

前記アプリケーション生成部が前記アプリケーションの生成に失敗した場合、

前記チャット制御部は、前記アプリケーションの生成の失敗により取得したエラーメッセージ、及び、再度、受信した該チャットメッセージを、前記生成 A I システムに送信し、前記アプリケーションの生成に係る要求を受信する、請求項 1 ～ 6 のいずれか 1 項に記載の情報処理装置。

[付記 8]

前記アプリケーション生成部が前記アプリケーションの生成に、一定回数、失敗した場合、

前記チャット制御部は、前記アプリケーションの生成の失敗した旨を前記端末装置に送信する、請求項 1 ～ 6 のいずれか 1 項に記載の情報処理装置。

[付記 9]

前記アプリケーションが表にデータが登録されるアプリケーションである場合、

前記パラメータは前記表の 1 つ以上のフィールド名である、付記 1 ～ 7 のいずれか 1 項に記載の情報処理装置。

[付記 10]

前記情報処理装置はネットワークを介して前記アプリケーションを実行するアプリケーションサービスと通信し、

前記アプリケーション生成部は、前記パラメータを指定して、前記アプリケーションサービスに対し、前記アプリケーションの生成を要求する、付記 1 ～ 8 のいずれか 1 項に記載の情報処理装置。

[付記 11]

前記端末装置では、前記アプリケーションサービスと通信する第一の W e b ブラウザと、前記第一の W e b ブラウザが起動させた第二の W e b ブラウザとが実行され、

前記情報処理装置は、前記第二の W e b ブラウザから前記チャットメッセージを受信し、

前記チャット制御部が、前記生成 A I システムが送信した前記アプリケーションの生成に係る要求を受信すると、

前記アプリケーション生成部は、前記第二の W e b ブラウザに前記アプリケーションの生成に係る要求を送信し、

前記第二の W e b ブラウザは前記第一の W e b ブラウザに前記アプリケーションの生成に係る要求を送信し、

前記第一の W e b ブラウザが前記アプリケーションサービスに対し、前記アプリケーションの生成に係る要求を送信する、付記 10 に記載の情報処理装置。

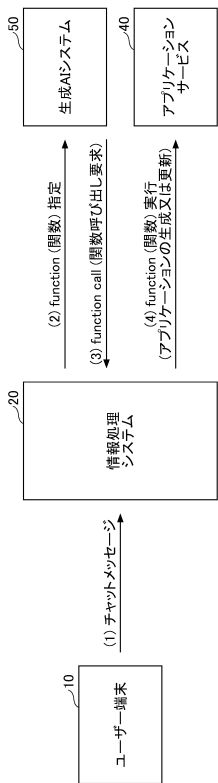
【符号の説明】

【0179】

- 10 ユーザー端末
- 20 情報処理システム
- 40 アプリケーションサービス
- 50 生成 A I システム
- 100 アプリ生成システム

【先行技術文献】

【特許文献】
【0180】
【特許文献1】特開2023-018851号公報
【図面】
【図1】



【図2A】

アプリ作成支援

🔍

どんなアプリを作ることができますか？ ～301

さまざまな業務に対応したアプリを作成することができます。具体的な例としては、以下のようなアプリが考えられます。

📋

1. 勤怠管理アプリ: 出勤・退勤時刻の記録や休暇申請など、従業員の勤怠情報を管理するアプリです。

2. 顧客管理アプリ: 顧客情報や営業活動の履歴を管理するアプリです。顧客の連絡先や商談の進捗状況などを管理することができます。

3. 問い合わせ管理アプリ: お客様からの問い合わせやサポート依頼を管理するアプリです。問い合わせ内容や対応状況を記録し、効率的な対応を行うことができます。

4. 在庫管理アプリ: 商品の在庫情報や入出庫履歴を管理するアプリです。在庫数の管理や発注の自動化など、効率的な在庫管理を行うことができます。

5. プロジェクト管理アプリ: プロジェクトの進捗状況やタスクの管理を行うアプリです。メンバーのタスク割り当てや進捗報告などを管理することができます。

これらは一部の例ですが、アプリケーションソフトでは様々な業務に対応したアプリを作成することができます。ユーザーの業務に合わせて必要なフィールドや機能を追加することができますので、具体的な要件に応じてアプリを作成することが可能です。

🔍

飲食店向けの在庫管理アプリを作ってみて下さい ～303

了解しました。飲食店向けの在庫管理アプリを作成します。

まずは、アプリのフィールド定義を決めるために、以下の質問にお答えください。

📋

1. 在庫アイテムの名称は何ですか？

2. 在庫アイテムの数量は数値で管理しますか？

3. 在庫アイテムの単位は何ですか？

4. 在庫アイテムのカテゴリはありますか？

5. 在庫アイテムの賞味期限は管理しますか？

6. 在庫アイテムの画像を添付する必要はありますか？

回答をお待ちしています。

🔍

適当に作ってみて下さい ～305

飲食店向けの在庫管理アプリを作成しました。以下のURLからアプリにアクセスすることができます。

飲食店在庫管理アプリ ～306a

このアプリでは、以下のフィールドが設定されています。

📋

・アイテム名: 在庫アイテムの名称を入力するフィールドです。

・数量: 在庫アイテムの数量を数値で管理するフィールドです。

・単位: 在庫アイテムの単位を入力するフィールドです。

・カテゴリ: 在庫アイテムのカテゴリを選択するドロップダウンフィールドです。食材、飲料、調味料、その他のいずれかを選択できます。

・賞味期限: 在庫アイテムの賞味期限を入力する日付フィールドです。

・画像: 在庫アイテムの画像を添付するためのフィールドです。

ご利用いただけるURLは上記の通りです。アプリにアクセスして在庫の管理を始めることができます。必要に応じてフィールドの追加修正を行うことも可能ですので、お気軽にご相談ください。

他にどんな入力項目を追加することが考えられますか？

このアプリをお客様に提案するにはどうすればいいと思いますか？

Send a message

【図2B】

キャンセル 保存 ～291

アイテム名 ～292

数量 ～293 単位

カテゴリ ～294

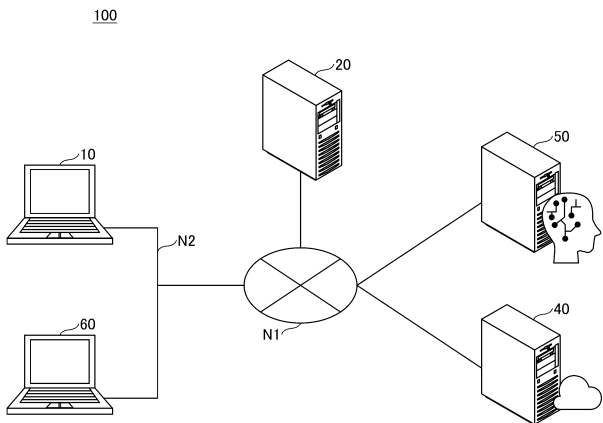
賞味期限 ～295

画像 ～296

ファイルを選択して下さい

ファイル選択

【図3】



10

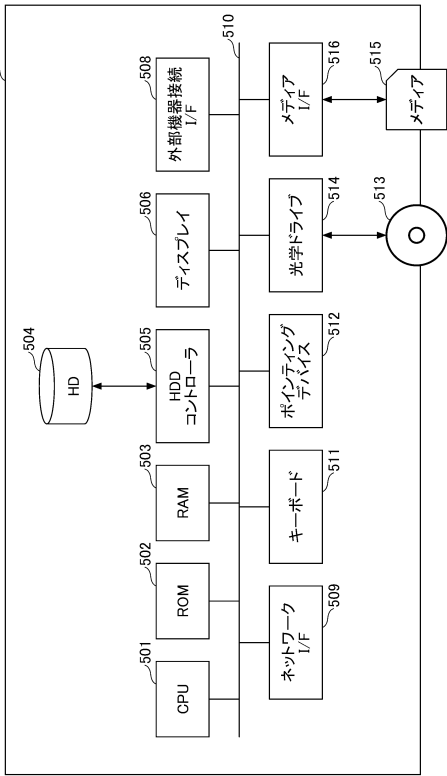
20

30

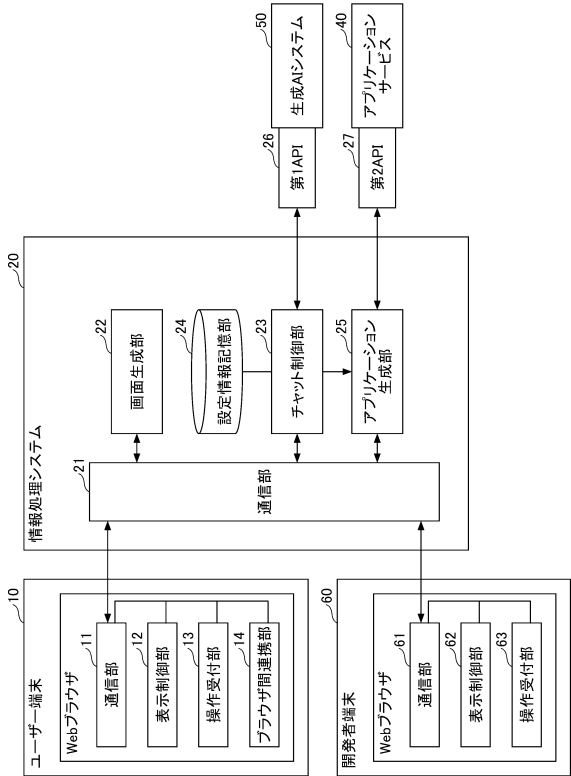
40

50

【図 4】



【図 5】



【図 6】

```

{
  "model": "gpt-3.5-turbo",
  "messages": [
    {
      "role": "system",
      "content": "あなたはこれから〇〇アプリケーションサービスと呼ばれる SaaS サービス内のアプリ開発を支援してもらいます。あなたにはこれから利用者と対話してアプリケーションの「フィールド」を決めてもらいます。フィールドの定義はJSON形式で表現します。例えば以下のような3つのフィールドを持つ日報アプリが考えられます:
      "
    },
    {
      "role": "user",
      "content": "日報アプリを作って下さい"
    }
  ],
  "functions": [
    {
      "name": "create_or_update_app",
      "description": "日報アプリを作成します。成功時は {\"ok\": true, \"value\": \"アプリの URL\"} を返し失敗時は {\"ok\": false, \"error\": \"...\"} を返します。",
      "parameters": {
        "type": "object",
        "required": [
          "properties",
          "name"
        ],
        "properties": {
          "name": {
            "type": "string"
          },
          "properties": {
            "type": "array",
            "items": {
              "required": [
                "code",
                "type",
                "label"
              ],
              "type": "object",
              "properties": {
                "code": {
                  "type": "string"
                },
                "type": {
                  "type": "string"
                },
                "label": {
                  "type": "string"
                }
              }
            }
          }
        }
      }
    }
  ]
}

```

【図 7】

```

{
  "choices": [
    {
      "message": {
        "role": "assistant",
        "content": null,
        "function_call": {
          "name": "create_or_update_app",
          "arguments": {
            "name": "日報",
            "properties": [
              {
                "code": "Date",
                "label": "日付",
                "type": "DATE"
              },
              {
                "code": "Content",
                "label": "業務内容",
                "type": "MULTILINE_TEXT"
              },
              {
                "code": "Impression",
                "label": "所感、学び",
                "type": "MULTILINE_TEXT"
              }
            ]
          }
        }
      }
    }
  ]
}

```

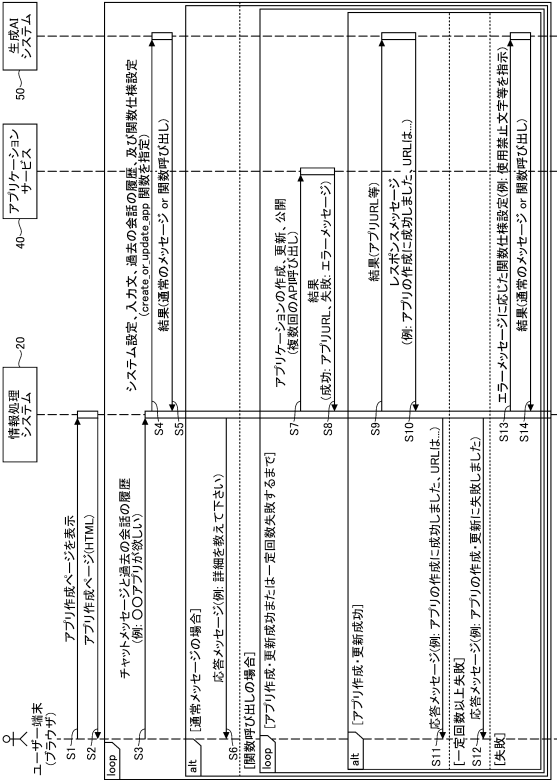
【図 8】

```
...json
{
  "model": "gpt-3.5-turbo",
  "messages": [
    {
      "role": "system", "content": "あなたは今から OOAアプリケーションサービスと呼ばれる SaaS サービスのアプリケーション開発を支援してもらいます (例)。" ~229
    },
    {
      "role": "user", "content": "日報アプリを作ってください" ~231
    },
    {
      "role": "assistant", "content": null, "function_call": { "name": "create_or_update_app", "arguments": { "app": {
        "role": "function", ~243a
        "name": "create_or_update_app", ~243b
        "content": { "url": "https://sample.com/k/42/*"} ~243c
      }
    }
  },
  "functions": [
    {
      "name": "create_or_update_app", ~234
      "description": "日報アプリを作成・更新する。2回目以降の呼び出しは更新", ~235
      "parameters": { ~236
        // アプリケーションサービスの API を呼び出すために必要な情報 (例)
      }
    }
  ],
  ...
}
```

【図 9】

```
...json
{
  "choices": [
    {
      "index": 0,
      "message": { "role": "assistant", "content": "アプリが作成されました。*"https://sample.com/k/42/*" から確認して下さい。" ~245
    },
    {
      "finish_reason": "stop"
    }
  ],
  // 省略
}
```

【図 10】



【図 11】

310

アプリ作成支援

顧客管理アプリを適当に作って下さい ~311

顧客管理アプリを作成しました。以下のURLからアクセスすることができます。

顧客管理アプリ

作成したアプリには以下のフィールドが含まれています:

顧客名

メールアドレス

電話番号

住所

会社名

業種

ステータス

メモ

ご確認ください。

他にどんな入力項目を追加することが考えられますか? ~313

顧客管理アプリに追加できる入力項目はさまざまです。以下にいくつかの例を挙げます。

誕生日: 顧客の誕生日を管理するためのフィールドです。

担当者: 顧客を担当する社員を選択するためのフィールドです。

売上予測: 顧客の売上予測を管理するためのフィールドです。

最終連絡日: 顧客との最終連絡日を管理するためのフィールドです。

重要度: 顧客の重要度を管理するためのフィールドです。

これらのフィールドを追加することで、より詳細な顧客情報を管理することができます。どのフィールドを追加したいか、または他に追加したい項目があれば教えてください。

担当者と重要度を追加して? ~315

顧客管理アプリに「担当者」と「重要度」のフィールドを追加しました。以下のURLからアクセスすることができます:

顧客管理アプリ ~317

アプリにアクセスして、フィールドの追加やデータの入力を行ってください。追加の要望や質問があればお知らせください。

他にどんな入力項目を追加することが考えられますか?

Send a message

312

314

316

10

20

30

40

50

【圖 1 2】

260

キャンセル

保存

〜271

顧客名

〜261

メールアドレス

〜262

@

電話番号

〜263

削除

住所

〜264

会社名

〜265

業種

〜266

ステータス

〜267

メモ

〜268

担当者

〜269

重要度

〜270

☒ 大

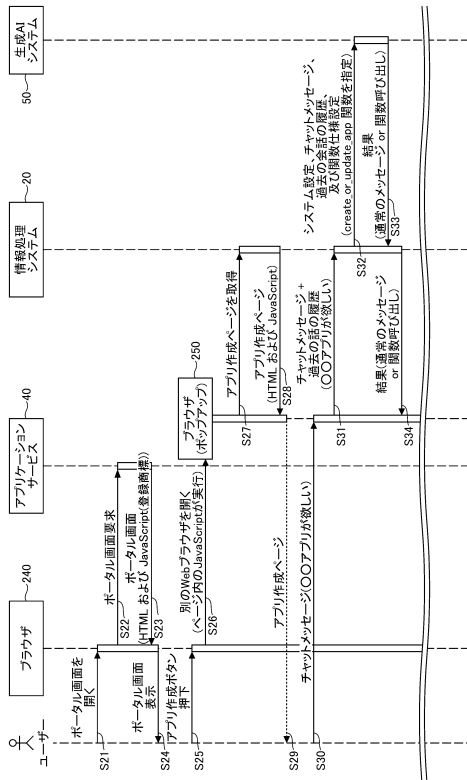
☐ 中

☐ 小

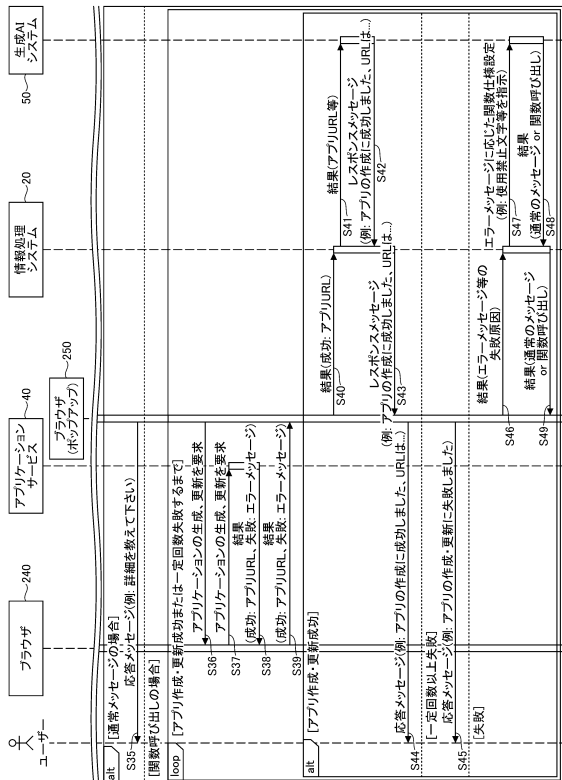
【圖 13】

[illegible]

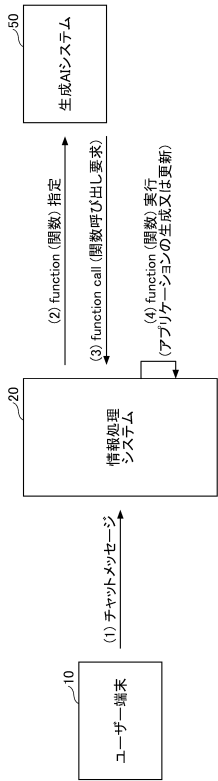
【図 14】



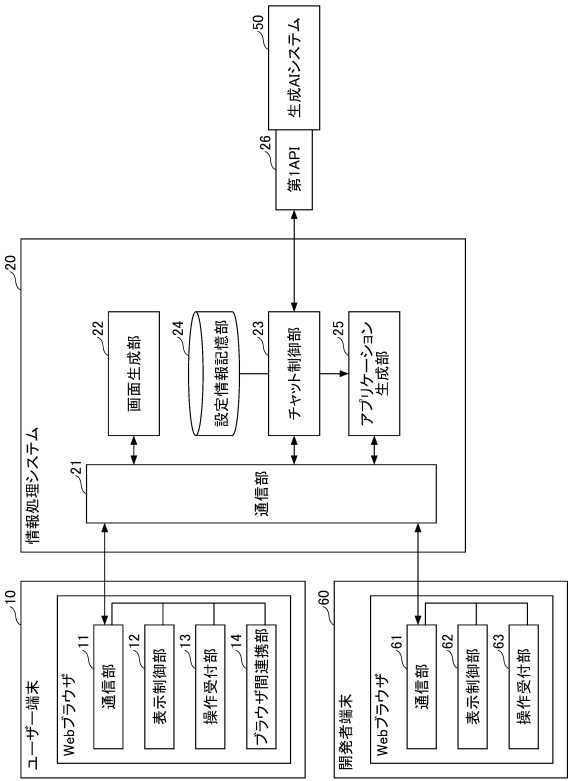
【図 15】



【図 16】



【図 17】



10

20

30

40

50