

(19) 日本国特許庁(JP)

(12) 特許公報(B2)

(11) 特許番号

特許第6165922号
(P6165922)

(45) 発行日 平成29年7月19日(2017.7.19)

(24) 登録日 平成29年6月30日(2017.6.30)

(51) Int.Cl. F I
G06F 9/44 (2006.01) G O 6 F 9/06 6 2 O K
G06F 11/36 (2006.01) G O 6 F 11/36 1 6 4

請求項の数 20 (全 27 頁)

(21) 出願番号	特願2016-95944 (P2016-95944)	(73) 特許権者	502208397
(22) 出願日	平成28年5月12日 (2016.5.12)		グーグル インコーポレイテッド
(62) 分割の表示	特願2013-511363 (P2013-511363) の分割		アメリカ合衆国 カリフォルニア州 94 043 マウンテン ビュー アンフィシ アター パークウェイ 1600
原出願日	平成23年5月19日 (2011.5.19)	(74) 代理人	110001195
(65) 公開番号	特開2016-173844 (P2016-173844A)		特許業務法人深見特許事務所
(43) 公開日	平成28年9月29日 (2016.9.29)	(72) 発明者	ヤセク・スラッシュスキー
審査請求日	平成28年6月10日 (2016.6.10)		ポーランド・30-087・クラクフ・ウ リツァ・リドラ・45ア/35
(31) 優先権主張番号	61/346,453	(72) 発明者	ジェイソン・ビー・パークス
(32) 優先日	平成22年5月19日 (2010.5.19)		アメリカ合衆国・テキサス・78703・ オースティン・パルマ・プラザ・1609
(33) 優先権主張国	米国 (US)		

最終頁に続く

(54) 【発明の名称】 バグクリアリングハウス

(57) 【特許請求の範囲】

【請求項1】

コンピュータで実装される方法であって、

アプリケーション開発者の第1のアプリケーションをオンラインアプリケーションストア上で分散できるようにするための第1の要求を、前記アプリケーション開発者のコンピューティングシステムから、かつ前記オンラインアプリケーションストアのコンピューティングシステムに提供することを含み、前記第1の要求は、(i)前記アプリケーション開発者を特定する情報と(ii)前記第1のアプリケーションのためのコンピュータコードとを含み、前記オンラインアプリケーションストアは、複数の独立している開発者からのアプリケーションを、前記複数の独立しているアプリケーション開発者から独立しているエンドユーザのコンピューティングデバイスに分散できるようにし、さらに、

前記オンラインアプリケーションストアが前記アプリケーション開発者の代わりに分散できるようにした1つ以上のアプリケーションに関する情報についての第2の要求を、前記アプリケーション開発者の前記コンピューティングシステムから、かつ前記オンラインアプリケーションストアの前記コンピューティングシステムに、後で提供することと、

前記オンラインアプリケーションストアが前記アプリケーション開発者の代わりに分散できるようにした1つ以上のアプリケーションに関する情報を、前記第2の要求に対する応答として前記アプリケーション開発者の前記コンピューティングシステムにおいて受信することとを含み、前記受信された情報は、1人以上のエンドユーザのコンピューティングデバイスが前記第1のアプリケーションの動作に関連して前記オンラインアプリケーシ

10

20

ョンストアに報告した問題に関する情報を含み、さらに、

前記1人以上のエンドユーザの前記コンピューティングデバイスが前記第1のアプリケーションの動作に関連して前記オンラインアプリケーションストアに報告した前記問題の少なくとも一部分を示す報告を、前記アプリケーション開発者の前記コンピューティングシステムにおいて表示することを含む、コンピュータで実装される方法。

【請求項2】

前記オンラインアプリケーションストアが前記アプリケーション開発者の代わりに分散できるようにした、前記第1のアプリケーションとは異なる1つ以上の追加的なアプリケーションの動作に関連して1人以上のエンドユーザのコンピューティングデバイスが、前記オンラインアプリケーションストアに報告した問題に関する追加的な情報を、前記アプリケーション開発者の前記コンピューティングシステムにおいて、かつ前記第2の要求に対する応答の一部として受信することを含む、

10

前記報告を表示することは、前記第1のアプリケーションおよび前記1つ以上の追加的なアプリケーションのうちの複数のアプリケーションの動作に関連して1人以上のエンドユーザのコンピューティングデバイスが前記オンラインアプリケーションストアに報告した問題に関する情報を表示することを含む、請求項1に記載のコンピュータで実装される方法。

【請求項3】

前記第1のアプリケーションは、携帯コンピューティングデバイス上で実行されるように構成される、請求項1に記載のコンピュータで実装される方法。

20

【請求項4】

前記1人以上のエンドユーザの前記コンピューティングデバイスが前記オンラインアプリケーションストアに報告した前記問題の前記少なくとも一部分を示す前記報告は、複数の問題の種類うちのそれぞれの問題の種類ごとに、前記1人以上のエンドユーザの前記コンピューティングデバイスが規定された時間間隔以内に前記オンラインアプリケーションストアに報告した前記それぞれの問題の種類に関連した問題の頻度を特定する、請求項1に記載のコンピュータで実装される方法。

【請求項5】

前記報告は、前記複数の問題の種類うちのそれぞれの問題の種類ごとに、

(i) 前記1人以上のエンドユーザの前記コンピューティングデバイスが最近の時間間隔以内に前記オンラインアプリケーションストアに報告した前記それぞれの問題の種類に関連した問題の頻度と、

30

(ii) 前記1人以上のエンドユーザの前記コンピューティングデバイスが前記最近の時間間隔前に生じた直前の時間間隔以内に前記オンラインアプリケーションストアに報告した前記それぞれの問題の種類に関連した問題の頻度とを特定する、請求項4に記載のコンピュータで実装される方法。

【請求項6】

前記1人以上のエンドユーザの前記コンピューティングデバイスが前記オンラインアプリケーションストアに報告した前記問題の前記少なくとも一部分を示す前記報告は、前記問題の前記少なくとも一部分のうちの各問題を、(i) 前記1人以上のエンドユーザの前記コンピューティングデバイスにおいてアプリケーションをクラッシュさせた問題についての第1の問題のグループと、(ii) 前記1人以上のエンドユーザの前記コンピューティングデバイスをフリーズさせた問題についての第2の問題のグループとに分類する、請求項1に記載のコンピュータで実装される方法。

40

【請求項7】

前記1人以上のエンドユーザの前記コンピューティングデバイスが前記オンラインアプリケーションストアに報告した前記問題の前記少なくとも一部分を示す前記報告は、1つ以上のプラットフォームのうちのそれぞれのプラットフォームごとに、前記それぞれのプラットフォーム上で生じたことが報告される問題の頻度を特定する、請求項1に記載のコンピュータで実装される方法。

50

【請求項 8】

前記第 2 の要求に回答して受信された前記情報は、個々のエンドユーザについての情報を特定することを含まない、請求項 1 に記載のコンピュータで実装される方法。

【請求項 9】

前記オンラインアプリケーションストアの前記コンピューティングシステムに前記第 1 の要求を提供することは、前記第 1 のアプリケーションに関連付けられたメタデータを送信することを含み、前記メタデータは、前記第 1 のアプリケーションのタイトルと、指定の価格を前記第 1 のアプリケーションについて請求するようにとの前記オンラインアプリケーションストアへの命令と、前記第 1 のアプリケーションのテキスト記述と、エンドユーザのコンピューティングデバイス上にまたは前記オンラインアプリケーションストアの 10 サイト上に提示されるべき前記第 1 のアプリケーションのためのアイコンと、のうち少なくとも 1 つを含む、請求項 1 に記載のコンピュータで実装される方法。

【請求項 10】

前記 1 人以上のエンドユーザの前記コンピューティングデバイスが前記オンラインアプリケーションストアに報告した前記問題の前記少なくとも一部分を示す前記報告は、前記アプリケーション開発者の前記コンピューティングシステムにおけるウェブページに提示される、請求項 1 に記載のコンピュータで実装される方法。

【請求項 11】

1 つ以上のプロセッサによって実行されると動作の実行を引起す命令が格納された 1 つ以上の非一時的なコンピュータ読取可能な媒体であって、前記動作は、 20

アプリケーション開発者の第 1 のアプリケーションをオンラインアプリケーションストア上で分散できるようにするための第 1 の要求を、前記アプリケーション開発者のコンピューティングシステムから、かつ前記オンラインアプリケーションストアのコンピューティングシステムに提供することを含み、前記第 1 の要求は、(i) 前記アプリケーション開発者を特定する情報と (i i) 前記第 1 のアプリケーションのためのコンピュータコードとを含み、前記オンラインアプリケーションストアは、複数の独立している開発者からのアプリケーションを、前記複数の独立しているアプリケーション開発者から独立している 30 エンドユーザのコンピューティングデバイスに分散できるようにし、さらに、

前記オンラインアプリケーションストアが前記アプリケーション開発者の代わりに分散できるようにした 1 つ以上のアプリケーションに関する情報についての第 2 の要求を、前記アプリケーション開発者の前記コンピューティングシステムから、かつ前記オンラインアプリケーションストアの前記コンピューティングシステムに、後で提供することと、 30

前記オンラインアプリケーションストアが前記アプリケーション開発者の代わりに分散できるようにした 1 つ以上のアプリケーションに関する情報を、前記第 2 の要求に対する応答として前記アプリケーション開発者の前記コンピューティングシステムにおいて受信することを含み、前記受信された情報は、1 人以上のエンドユーザのコンピューティングデバイスが前記第 1 のアプリケーションの動作に関連して前記オンラインアプリケーションストアに報告した問題に関する情報を含み、さらに、

前記 1 人以上のエンドユーザの前記コンピューティングデバイスが前記第 1 のアプリケーションの動作に関連して前記オンラインアプリケーションストアに報告した前記問題の 40 少なくとも一部分を示す報告を、前記アプリケーション開発者の前記コンピューティングシステムにおいて表示することを含む、1 つ以上の非一時的なコンピュータ読取可能な媒体。

【請求項 12】

前記動作は、

前記オンラインアプリケーションストアが前記アプリケーション開発者の代わりに分散できるようにした、前記第 1 のアプリケーションとは異なる 1 つ以上の追加的なアプリケーションの動作に関連して 1 人以上のエンドユーザのコンピューティングデバイスが前記オンラインアプリケーションストアに報告した問題に関する追加的な情報を、前記アプリケーション開発者の前記コンピューティングシステムにおいて、かつ前記第 2 の要求に対 50

する応答の一部として受信することを含み、

前記報告を表示することは、前記第1のアプリケーションおよび前記1つ以上の追加的なアプリケーションのうちの複数のアプリケーションの動作に関連して1人以上のエンドユーザのコンピューティングデバイスが前記オンラインアプリケーションストアに報告した問題に関する情報を表示することを含む、請求項11に記載の1つ以上の非一時的なコンピュータ読取可能な媒体。

【請求項13】

前記第1のアプリケーションは、携帯コンピューティングデバイス上で実行されるように構成される、請求項11に記載の1つ以上の非一時的なコンピュータ読取可能な媒体。

【請求項14】

前記1人以上のエンドユーザの前記コンピューティングデバイスが前記オンラインアプリケーションストアに報告した前記問題の前記少なくとも一部分を示す前記報告は、複数の問題の種類の中のそれぞれの問題の種類ごとに、前記1人以上のエンドユーザの前記コンピューティングデバイスが規定された時間間隔以内に前記オンラインアプリケーションストアに報告した前記それぞれの問題の種類に関連した問題の頻度を特定する、請求項11に記載の1つ以上の非一時的なコンピュータ読取可能な媒体。

【請求項15】

前記報告は、前記複数の問題の種類の中のそれぞれの問題の種類ごとに、

(i) 前記1人以上のエンドユーザの前記コンピューティングデバイスが最近の時間間隔以内に前記オンラインアプリケーションストアに報告した前記それぞれの問題の種類に関連した問題の頻度と、

(ii) 前記1人以上のエンドユーザの前記コンピューティングデバイスが前記最近の時間間隔前に生じた直前の時間間隔以内に前記オンラインアプリケーションストアに報告した前記それぞれの問題の種類に関連した問題の頻度とを特定する、請求項14に記載の1つ以上の非一時的なコンピュータ読取可能な媒体。

【請求項16】

前記1人以上のエンドユーザの前記コンピューティングデバイスが前記オンラインアプリケーションストアに報告した前記問題の前記少なくとも一部分を示す前記報告は、前記問題の前記少なくとも一部分の中の各問題を、(i) 前記1人以上のエンドユーザの前記コンピューティングデバイスにおいてアプリケーションをクラッシュさせた問題についての第1の問題のグループと、(ii) 前記1人以上のエンドユーザの前記コンピューティングデバイスをフリーズさせた問題についての第2の問題のグループとに分類する、請求項11に記載の1つ以上の非一時的なコンピュータ読取可能な媒体。

【請求項17】

前記1人以上のエンドユーザの前記コンピューティングデバイスが前記オンラインアプリケーションストアに報告した前記問題の前記少なくとも一部分を示す前記報告は、1つ以上のプラットフォームの中のそれぞれのプラットフォームごとに、前記それぞれのプラットフォーム上で生じたことが報告される問題の頻度を特定する、請求項11に記載の1つ以上の非一時的なコンピュータ読取可能な媒体。

【請求項18】

前記第2の要求に応答して受信された前記情報は、個々のエンドユーザについての情報を特定することを含まない、請求項11に記載の1つ以上の非一時的なコンピュータ読取可能な媒体。

【請求項19】

前記オンラインアプリケーションストアの前記コンピューティングシステムに前記第1の要求を提供することは、前記第1のアプリケーションに関連付けられたメタデータを送信することを含み、前記メタデータは、前記第1のアプリケーションのタイトルと、指定の価格を前記第1のアプリケーションについて請求するようにとの前記オンラインアプリケーションストアへの命令と、前記第1のアプリケーションのテキスト記述と、エンドユーザのコンピューティングデバイス上にまたは前記オンラインアプリケーションストアの

10

20

30

40

50

サイト上に提示されるべき前記第1のアプリケーションのためのアイコンと、のうち少なくとも1つを含む、請求項11に記載の1つ以上の非一時的なコンピュータ読取可能な媒体。

【請求項20】

第1のコンピューティングシステムであって、

1つ以上のプロセッサと、

1つ以上のプロセッサによって実行されると動作の実行を引起す命令が格納された1つ以上の非一時的なコンピュータ読取可能な媒体とを備え、前記動作は、

アプリケーション開発者の第1のアプリケーションをオンラインアプリケーションストア上で分散できるようにするための第1の要求を、前記アプリケーション開発者の前記第1のコンピューティングシステムから、かつ前記オンラインアプリケーションストアのコンピューティングシステムに提供することを含み、前記第1の要求は、(i)前記アプリケーション開発者を特定する情報と(ii)前記第1のアプリケーションのためのコンピュータコードとを含み、前記オンラインアプリケーションストアは、複数の独立している開発者からのアプリケーションを前記複数の独立しているアプリケーション開発者から独立しているエンドユーザのコンピューティングデバイスに分散できるようにし、さらに、

前記オンラインアプリケーションストアが前記アプリケーション開発者の代わりに分散できるようにした1つ以上のアプリケーションに関する情報についての第2の要求を、前記アプリケーション開発者の前記第1のコンピューティングシステムから、かつ前記オンラインアプリケーションストアの前記コンピューティングシステムに、後で提供することと、

前記オンラインアプリケーションストアが前記アプリケーション開発者の代わりに分散できるようにした1つ以上のアプリケーションに関する情報を、前記第2の要求に対する応答として前記アプリケーション開発者の前記第1のコンピューティングシステムにおいて受信することを含み、前記受信された情報は、1人以上のエンドユーザのコンピューティングデバイスが前記第1のアプリケーションの動作に関連して前記オンラインアプリケーションストアに報告した問題に関する情報を含み、さらに、

前記1人以上のエンドユーザの前記コンピューティングデバイスが前記第1のアプリケーションの動作に関連して前記オンラインアプリケーションストアに報告した前記問題の少なくとも一部分を示す報告を、前記アプリケーション開発者の前記第1のコンピューティングシステムにおいて表示することを含む、第1のコンピューティングシステム。

【発明の詳細な説明】

【技術分野】

【0001】

この出願は、2010年5月19日に提出された米国仮出願シリアルNo. 61/346453、発明の名称が“バグクリアリングハウス”の優先権を主張し、その全内容が参照によってここに組み込まれる。

【0002】

この出願は、ソフトウェアアプリケーションに対する問題に関する情報の管理および分散に関する。

【背景技術】

【0003】

ソフトウェア開発は、困難なビジネスである。将来のソフトウェア開発者は先ず、ソフトウェアのアーキテクチャリングおよびプログラミングについて訓練する必要がある。その後、絶え間なく変わるプログラミング言語および技術を最新のものに維持する必要がある。そして、アイデアを有する必要がある。開発者は次いで、対象のコンピューティングプラットフォーム上で利用可能なリソース内で、および自分の時間ないしは資金的制約内で、プログラミングを介してどのようにアイデアを実装するかを考え出す必要がある。また、異なる設定でプログラムを人々に対して検証して、プログラムのバグを修正するように訂正を行う必要がある。次いで、開発者はどのように、ソフトウェアの値段を付けて、

10

20

30

40

50

分散 (distribute) して、人々にその興味を持たせるかを考え出す必要がある。

【 0 0 0 4 】

高度なプログラミング環境は、これらの問題の一部を解決するのに役立つ、即ち組織を設立し、コードのタイピングをより自動化することである。また、スマートフォンおよびネットブックのようなワイヤレスコンピューティングデバイスに関するオンラインアプリケーションストアの最近の到来は、規模の小さい開発者にとってより簡単にソフトウェアを市場に出して販売することを可能にした。これらの開発の両方によって、ソフトウェア開発者およびそのソフトウェアを購入する人々の生活をより良いものにした。一方で、ソフトウェア開発者の生活が辛い一面もある。例えば、プログラムのユーザからフィードバックを取得して、効果的な方法でそのフィードバックを分析するのが困難なことがある。例えば、多数の開発者は、消費者がフィードバックを提供できるアプリケーションを電子メールリンクに提供する。

10

【 発明の概要 】

【 発明が解決しようとする課題 】

【 0 0 0 5 】

本明細書は、ソフトウェア開発者に対して、特にスマートフォン等のワイヤレス携帯コンピューティングデバイスのユーザのための第三者オンラインアプリケーションストアを介してアプリケーションを販売および分散するソフトウェア開発者に対して、問題、即ちバグの報告を管理するためのシステムおよび技法を検討する。例えば、開発者は、一般のアプリケーションストアを提供するシステムで登録される個人のプログラマでもよく、それにより開発者は、全て精通した周知の方法で、アプリケーションストアから購入を行うために登録されている一般のメンバーに対するアプリケーションの分散のために、アプリケーションストアにアプリケーションをアップロードできる。特定のソフトウェアアプリケーション、および/または1つまたは複数の携帯コンピューティングデバイス上で実行するオペレーティングシステムは、問題、即ちバグ(例えば、アプリケーションクラッシュまたはフリーズ、過度の電力消費量、またはデバイスのユーザがアプリケーションを改善するための問題またはアイデアを手動で報告すること)がある時、情報を収集するようにプログラムされてもよく、アプリケーションストアに関連付けられ、アプリケーションストアを運営する同一の組織によって経営される報告サブシステムに、問題に関するデータを送信してもよい。

20

30

【 0 0 0 6 】

報告サブシステムは、(デバイスによって自動的に生成されるか、またはデバイスのユーザによって手動で生成される)そのような入力バグ報告の提供の各々を、問題の対象になったコンピュータアプリケーション(例えば、開かれたアプリケーションおよび/またはフリーズまたはクラッシュが生じた時のオペレーティングシステムの焦点)に関連付けることができ、問題の特徴付ける情報を格納できる。情報は、アプリケーションに対して、および拡張によりアプリケーションの開発者に対して関連付けられるような方法で格納されてもよい。開発者(アプリケーションストアにソフトウェアを提供した個人またはその個人のエージェントでもよい)は、次にシステムにログオンする時、開発者は、自分がアップロードしたソフトウェアアプリケーションに関する報告を引き出すことができ、ここで報告は、そのようなソフトウェアアプリケーションの各々に関する全ての報告された問題に対する全データを集約し、またアプリケーションに関する問題の種類および程度を示す。

40

【 0 0 0 7 】

そのような報告は、例えば携帯コンピューティングデバイスが問題を経験して対応するデータを生成したユーザに関するプライバシーを維持するために、報告サブシステムによってそこから除去された特定の情報を有することがある。サブシステムによって開発者に提供される情報のレベルはまた、サブシステムが開発者に対して有する信用のレベルに依存することができる。また、ユーザは、共有されうる情報のレベルに関して通知されてもよく、そのような情報の使用を許可または不許可するかを選択してもよい。

50

【0008】

報告は、アプリケーションまたは複数のアプリケーションに関する問題の発生の頻度を示す要約報告、アプリケーションに関する特定タイプの全問題をリストし、問題（例えば、どのコード部分に問題が生じたか）を記述するデータを提供するエラー報告、および問題がデバイス上で生じた時に各コンピューティングデバイスの状態に関する情報を示すスタックトレース等、各種精通した形式を取ってもよい。

【0009】

また、システムは、多数のアプリケーションにわたって第三者プロバイダにより広く役立つデータを生成してもよい。例えば、アプリケーションストアを運営する組織は、アプリケーションストアにアクセスする携帯デバイスにオペレーティングシステムを提供し、多数のアプリケーションに関する問題データは、各種方法で“ロールアップ（roll up）”されて集約データおよび結果報告を生成し、オペレーティングシステムに関する問題があるかを判断してもよい。例えば、そのようなデータは、特定のオペレーティングシステムまたはインタフェースにアクセスする時に多数の異なるアプリケーションがフリーズを経験していることを示す場合、第三者プロバイダは、サービスまたはインタフェースに関するコードに問題の可能性のあることを警告されうる。

【0010】

そのようなシステムおよび技法は、特定の実施形態で1つまたは複数の利点を提供しうる。例えば、開発者は、単に特定の開発環境を使用して、自分の持つバグ追跡システムを設定する高度な会社にものみ利用可能な診断機能を備えてよい。デバイスのユーザは、各種開発者から改善されたアプリケーションを受信でき、多数の異なる開発者からの多数の異なるアプリケーションにわたって一貫している都合の良い方法でバグ報告を可能にする。そしてオペレーティングシステムおよび/または関連アプリケーションストアのようなプラットフォームのプロバイダは、開発者およびデバイスのユーザに対するそのプラットフォームの望ましさ（desirability）を改善しうる。

【0011】

1つの実施形態では、ソフトウェア問題報告を管理するためのコンピュータ実装されたシステムが検討される。システムは、多数の異なる開発者組織からのソフトウェア開発者を登録するために登録サブシステムを含む。システムはまた、複数のアプリケーション開発者からの複数のアプリケーションを一般のメンバーが獲得できるようにするアプリケーションストアを含む。システムはまた、アプリケーションストアを使用して分散されたアプリケーションに関する問題の報告を受信し、問題に関するデータを受信し、特定の開発者または特定のアプリケーションの開発者にデータを関連付けるようにプログラムされたアプリケーションバグ追跡装置を含む。また、システムは、1つまたは複数のアプリケーションを提供した開発者に関する1つまたは複数の問題報告を生成するために報告生成装置を含み、問題報告は、開発者によってアプリケーションストアに提供された特定のアプリケーションに関する問題に関するデータについての情報を含む。

【0012】

実施形態は、以下の特徴の全てを含むかまたは含まなくてもよい。問題の報告は、アプリケーションクラッシュ、アプリケーションフリーズ、過度のバッテリー使用、および手動で起動されたコンピューティングデバイスのユーザコメントからなるグループから選択された問題を含むことができる。システムはまた、問題の報告が特定のコンピューティングデバイスモデルまたはコンピュータオペレーティングシステムに関する問題を示すか否かを特定するために多数のアプリケーションにわたって問題の報告を分析するようにプログラムされたバグ分析装置を含むことができる。バグ分析装置はまた、特定のデバイスモデルまたはオペレーティングシステムバージョンに関して統計的に比較的高い報告頻度を特定することによって、問題の報告が特定のコンピューティングデバイスモデルまたはコンピュータオペレーティングシステムに関する問題を示すことを特定できる。システムは、開発者が取得する権利を有する特定のアプリケーションのユーザに関する情報がどれくらいあるかを特定する信用レベルに基づき、開発者に提供される情報を制限するように配列

10

20

30

40

50

されうる。システムはまた、開発者が、特定の問題が生じた装置に対するオペレーティングシステムを管理する組織からのコアアプリケーションの開発者であるか否かに基づき、開発者に提供される情報を制限するようにプログラムされうる。

【0013】

また、システムは、問題のタイプに従って、問題に関する情報をグループ化するように配列されうる。システムは、特定の問題が生じたデバイスに対するオペレーティングシステム識別子およびデバイスタイプを示す情報を提供するように配列されうる。システムはまた、開発者からのアプリケーションのアップロードを受け入れ、アプリケーションがアップロードされた開発者アカウントに各アプリケーションを関連付ける、開発者アップロードインタフェースを含みうる。システムは、特定の開発者に対する異なるバグの割合から判断される深刻レベルによってソートされるバグを、開発者に提供するようにプログラムされうる。

10

【0014】

1つの局面では、ソフトウェア問題を管理するためのコンピュータ実装された方法は、中央サーバシステムから離れた複数の異なるコンピューティングデバイスから複数のソフトウェアアプリケーション問題報告を中央サーバシステムで受信することを含む。システムはまた、特定のソフトウェア開発者によって提供された特定のアプリケーションに、問題報告の特定の1つを関連付けることを含む。また、システムは、特定の開発者の1人から識別情報を受信することを含む。システムはまた、識別情報の受信に応じて、1人の開発者によって中央サーバシステムで管理される特定のアプリケーションに関する問題報告を記述する情報を、特定の開発者の1人に提供することを含む。

20

【0015】

実施形態は、以下の特徴の全てを含むか、または含まなくてもよい。ソフトウェアアプリケーション問題報告は、フリーズ、クラッシュ、およびユーザにより生成されたコメントに関する報告を含みうる。また、ソフトウェアアプリケーション問題報告は、特定の問題報告を生成したデバイスに関するトレースデータ、およびデバイスの構成に関する情報を含みうる。システムはまた、複数のソフトウェア開発者からソフトウェアアプリケーションアップロードを受信し、アプリケーションに関する開発者アカウントに各アプリケーションを関連付けることを含む。

【図面の簡単な説明】

30

【0016】

【図1A】図1Aは、多数の開発者からのソフトウェアを分散し、ソフトウェアの動作に関する情報を集めるためのシステム100を示す概念図である。

【図1B】図1Bは、ソフトウェアアプリケーションの動作上のフィードバックを提供するための動作を示すフロー図である。

【図2】図2は、それらデバイス上のソフトウェアアプリケーションの動作に関するコンピューティングデバイスからの提供を追跡するためのシステム200の概略図である。

【図3】図3は、ソフトウェア問題報告を管理するためのプロセスのフローチャートである。

【図4】図4は、多数の開発者からのソフトウェアに関する問題の提供を管理するためのプロセスのスィムレーン図である。

40

【図5A】図5Aは、ソフトウェアアプリケーションに関して動作上の問題を受けた携帯コンピューティングデバイスの例示的なスクリーンショットである。

【図5B】図5Bは、ソフトウェアアプリケーションに関して動作上の問題を受けた携帯コンピューティングデバイスの例示的なスクリーンショットである。

【図6A】図6Aは、ソフトウェア追跡システムによって生成される開発者報告の例示的なスクリーンショットである。

【図6B】図6Bは、ソフトウェア追跡システムによって生成される開発者報告の例示的なスクリーンショットである。

【図6C】図6Cは、ソフトウェア追跡システムによって生成される開発者報告の例示的

50

なスクリーンショットである。

【図7】図7は、包括的なコンピュータデバイス700および包括的な携帯コンピュータデバイス750の例を示す。

【発明を実施するための形態】

【0017】

本明細書は、ソフトウェアアプリケーションの動作上のフィードバックを提供するために使用されうる各種技術および技法を開示する。具体的な例では、互いに独立している（例えば、同じ会社で働いていない、同じドメインから働いていない、または共通のソフトウェア開発システムを共有していない）ソフトウェア開発者の分散したグループは、コンピューティングデバイスの各種ユーザがアプリケーションをレビューおよびダウンロードし、しばしばアプリケーションを購入またはライセンスの料金を払うことができる（各開発者が支払いの割り当てを受け取る）アプリケーションストアにそのソフトウェアアプリケーションをアップロードすることがある。デバイス上のオペレーティングシステムは、スタックトレースを維持し、そのような情報を、デバイスに問題、即ちバグがある時、中央サーバシステムにそのような情報を報告することを含む、デバイス上の問題の一般的なエラー処理を実行することができる。そのような情報は、改善策を取ることができ、関連するソフトウェアコードが訂正および更新できるように、問題の原因を診断しようと試みる訓練された個人によって共通に使用される。

【0018】

ここに開示される例では、各種コンピューティングデバイスから提供されるデータの一部はまた、問題を起こしたか、または問題が生じた時にアクティブだったコードを提供した開発者がレビューできるように作成されうる。そのような動作は、問題が生じた時に関連するアプリケーションまたは複数のアプリケーションを特定し、問題に関する情報をアプリケーションまたは複数のアプリケーションを特定する情報とともに、中央サーバシステムに提供するオペレーティングシステムによって生じうる。中央サーバシステムは次いで、ソフトウェアアプリケーション識別子に関連してデータを格納でき、そのような識別子はまた、アプリケーションの開発者を特定するために使用されうる。（アプリケーションストアまたはオペレーティングシステムの経営者と比較して全く異なる個人または組織でありうる）そのような開発者がシステムにログインし、自分のコードに関する問題の報告を要求する時、システムは、問題データが格納されているデータベースに問合せて特定の開発者に関する全データを発見し、各種方法でデータをフィルタ処理してグループ化する1つまたは複数の報告を生成できる。一例として、個人的に特定可能な情報は、報告に提供されるデータから除去されうるか、または曖昧(obscrete)にされうる。また、データは、各アプリケーションに従って格納できるので、開発者は、多数のそのようなアプリケーションを有する場合に、自分の各アプリケーションの相対的な動作に関する理解を素早く得ることができる。

【0019】

図1Aは、多数の開発者からのソフトウェアを分散し、ソフトウェアの動作に関する情報を集めるためのシステム100を示す概念図である。一般に、システム100は、開発者のグループとユーザのグループ（勿論、開発者の一部がユーザでもよく、その逆でもよい）との間に位置する中央サーバシステムを含む。中央サーバシステムは、開発者からソフトウェアコードアップロードを受信し、ユーザにコードダウンロードを提供し、ユーザからエラーまたは問題データを受信し、開発者への提示のためにそのようなデータをフィルタ処理および整理(organize)する。

【0020】

図をさらに詳細に参照すると、中央サーバシステムは、アプリケーションセンター102の形式でここに示される。アプリケーションセンター102は、ユーザセンター108および開発者センター110をなす物理的構成要素および論理的構成要素を含む。そのような構成要素は例えば、それぞれユーザおよび開発者にサービスを提供するためのユーザインタフェースを生成するフロントエンドまたはウェブサーバを含むことができる。ユー

10

20

30

40

50

ザセンター 108 はこの例では、アプリケーションセンター 102 の経営者によって開発された特定のソフトウェアオペレーティングシステムを動作させる携帯電話またはアプリケーション電話 (app phone) を獲得しているそれぞれ個人であるユーザ 104 と通信する。オペレーティングシステムを備えた全サービスの一部として、経営者は、携帯デバイスに関するソフトウェアアプリケーションのレビューおよび購入にアプリケーションセンター 102 が利用できるようにする。故に例えば、ユーザ A (例えば、10 代) は、複数のビデオゲームをダウンロードした可能性があり、ユーザ B (例えば、女性社員) は、ホストされたクラウドベースのコンピューティングプラットフォームを利用する各種ビジネスアプリケーションをダウンロードした可能性があり、ユーザ C (例えば、大学生) は、(複数の電子書籍教科書と共に) 電子書籍リーダアプリケーションおよび各種テスト準備アプリケーションを、数学モデル化アプリケーションと共にダウンロードした可能性がある。各状況では、関連するユーザは、(自分の電話と共に出荷されたアプリケーションを補足しうる) 自分のダウンロードしたアプリケーションの各々を立ち上げ、類似の方法でそれと対話しうる。ユーザはまた、必要が生じる時、追加アプリケーションを発見してダウンロードするために、アプリケーションセンター 102 を再訪問することができる。

【0021】

時々、問題は、ユーザの 104 デバイスの各々上で生じることがある。例えば、ユーザは、複数のアプリケーションを同時に動作していることがあり、アプリケーションの 1 つにクラッシュを引起こすメモリオーバフロー条件を作り出すことがある。代わりに、アプリケーションは、それを閉じさせる動作を実行することにより、ユーザは、アプリケーションが実行しているプロセスを終了する必要がある。代わりに、デバイスのユーザは、問題提供プロセスを呼び出すことができ、それによりユーザは、アプリケーションのエラーを特定するために、またはアプリケーションに対して改善された機能を提供するために、フォームに書き込むことができる。これらのインスタンスおよび類似のインスタンスの各々は、バグとしてここに集められて記載され、その点では、フィードバックがアプリケーションの開発者に与えられる必要があるソフトウェアアプリケーションに関する問題を示す(但し、問題は結局、オペレーティングシステム、デバイスハードウェア、または別のアプリケーションに起因すると判断されることがある)。

【0022】

ここで示す通り、アプリケーションは、アプリケーションストア 112 から獲得でき、それは、(アプリケーションストア 112 の経営者によって提供される最上位アプリケーションの一部に追加して) 第三者によって提供されるアプリケーションが、ウェブページの一部として表示可能なウェブサイト等の類似の形式を取ることができ、そのウェブページ上には、ユーザによって選択された時、システム 100 でユーザのアカウントにログインされるデバイスに、特定の選択されたアプリケーションを供給またはダウンロードさせる制御が表示される。

【0023】

ユーザセンター 108 はまた、バグ収集装置 116 を含み、それは、直接、またはアプリケーションのインスタンスの動作に関して報告するオペレーティングシステム構成要素を介して、ユーザの 104 デバイスに分散されるアプリケーションのインスタンスと自動的に通信するための構成要素を含む。バグ収集装置はまた、提供の時間および日付、デバイスに関するオペレーティングシステムおよびバージョン、デバイスの製造およびモデル、問題イベントが生じた時にデバイス上で実行していた他のアプリケーション、最近のデバイスアクティビティのトレース、デバイスの構成データ、および他の類似の情報等の情報を含む 1 つまたは複数のデータベースにデータを配列する等により、永続ストレージ (persistent storage) に集められた情報を再フォーマットすることができる。

【0024】

アプリケーションセンター 102 の他の“サイド”には、開発者 106 が示され、それは、ユーザ 104 のように、かなり多数存在し、地理的に分布されてもよい。特に、ユーザ 104 および開発者 106 は、世界中に広がっていてもよく、インターネットを含む各

10

20

30

40

50

種ネットワークを介してアプリケーションセンター102にアクセスできる。一例として、システム100は、何千または何万ものアクティブな開発者106および何十万または何百万（または一億）ものアクティブなユーザ104を有することができる。ここで示す技法は特に、比較的大きなユーザのデータが各開発者に供給され、そのようなシステムの開発者は、比較的小規模であり、独自のバグ報告システムを開発できないような、大きな開発者の基盤を備えたシステムで、都合よく作動する。

【0025】

開発者センター110の第1のメイン構成要素は、アプリケーションマネージャ114である。この構成要素は、ユーザに関するアプリケーションストア112の反対側でもよい。特に、アプリケーションマネージャ114は、システム100の正当な開発者メンバーとして登録するために、および自分が開発したアプリケーションに関するコードまたは自分が商品流通を管理しているコードを後でアップロードするために、開発者がアクセスまたは対話しうるウェブサーバおよび関連するソフトウェアを含むことができる。アプリケーションマネージャ114はまた、複数の追加サービスを提供でき、それは一般に、開発者と対話することで周知であり、開発者が自分のアプリケーションを都合よく管理できるようにする。

【0026】

バグ報告装置118は、バグ収集装置116に対する開発者サイドのものとして機能し、（楽観的な開発者は、問題を機会としてみなすかもしれないが）問題がソフトウェアに関して生じる時、ソフトウェア分散を達成し、および開発者106で開始し、アプリケーションマネージャ114およびアプリケーションストア112に移動し、次いでアプリケーションを獲得および実行するユーザ104に移動し、バグ収集装置116、バグ報告装置118、および開発者106に戻るフィードバックサイクルを達成する。

【0027】

バグ報告装置118は特に、開発者106との対話のためのウェブサーバシステムと、各開発者に対応するアプリケーションに関して、ユーザの104デバイスによって経験された問題に関するデータを提供するための報告生成機能とを提供する。報告は、標準（standard）またはカスタムフォーマットでもよい。標準報告は、（ここでは単一のエンティティとして示されるが、互いに協働する多数の異なる組織によって経営されうる）アプリケーションセンター102の経営者によって開発されてもよく、どのように自分の各アプリケーションが実行しているかを見ている開発者によって簡単にアクセスおよび生成されてもよい。カスタム報告は、各個人の開発者によって形成されてもよく、周知のグラフィカル報告開発ツールを使用してもよい。例えば、開発者は、自分の報告で見たいデータの特定のフィールドを選択することがあり、自分の報告の視覚的様子を定義してもよい。開発者はまた、生データの形式で（報告行為がユーザ104間で生じた時にバッチ形式およびリアルタイムの両方で）報告を受信し、自分の機器を使用してデータに関して操作および報告してもよい。また、開発者106は、自分が見たい集約報告を定義してもよい。例えば、開発者が一式の製品（例えば、業績の一式）を開発した場合、開発者は、一式の各構成要素に共通なパラメータに関するセクションと、各構成要素および特定の構成要素に固有なデータに関する分離セクションとを含む報告を望むことができる。

【0028】

これらの方法では、システム100は、注意を必要としうる開発者のソフトウェア内のバグに関する拡張した役立つフォーマットデータを開発者106に提供できる。特定の開発者は、（オペレーティングシステムが、実行している時にアプリケーションを特定できるメカニズムを提供するのではなく、）特定のデータを得るために、自分のソフトウェアに関して何もしないでよい。特定のインスタンスでは、開発者は、APIを介して等、追加のデータを報告させてもよいが、再び、開発者は、バグ提供（bug submission）を受信するために、クライアントデバイス上のデータを収集するために、および全アプリケーションメンテナンスシステムをも実装するために、全システムを実装する必要がない。結果として、入門レベルの開発者は、自分のアプリケーションを改善するためにプログラムの

10

20

30

40

50

利益を得ることができ、アプリケーションの消費者は、同様に利益になる。

【 0 0 2 9 】

図 1 B は、ソフトウェアアプリケーションの動作上のフィードバックを提供するための動作を示すフロー図である。一般に、アプリケーション開発者が自分のソフトウェアアプリケーションを一般の人に利用可能にして、アプリケーションの動作上のフィードバックを受信し、それによりアプリケーションを容易に、および迅速に改善できる例を説明するために、グラフィカルな時系列がここで示され、左上の角で開始し、右下の方で終了する。

【 0 0 3 0 】

動作 1 2 0 では、アプリケーション開発者は、携帯アプリケーションマーケットにアプリケーションを提供する。提供は、各種適切な方法でもよく、開発者は、アプリケーションを正当化するのを助ける自己署名証明書をアプリケーションに提供し、システムにセキュリティを提供できる。動作 1 2 2 は、同じ開発者によって、または他の開発者によって先にアップロードされた可能性がある 3 つの他のアプリケーションと共に、携帯アプリケーションマーケット内のアプリケーションの存在を示す。アプリケーションの“所有権”は、アプリケーションをアップロードした、通常は実際の開発者か、または開発者の代わりに働く者でありうるユーザのアカウントにアプリケーションを関連付けることによって維持できる。

10

【 0 0 3 1 】

動作 1 2 4 では、携帯コンピューティングデバイス（例えば、スマートフォンまたはアプリケーション電話）のユーザは、携帯アプリケーションマーケットを訪問し、自分のアカウント下でシステムにログインすることにより自分自身を特定している。ユーザは次いで、動作 1 2 0 でアップロードされたアプリケーションを選択し、アプリケーションは、インストールされ、周知の方法で、携帯デバイスのユーザによる実行に備えている。

20

【 0 0 3 2 】

動作 1 2 6 では、爆発のアイコンは、アプリケーションが携帯デバイス上で動作している間、エラーの発生を示す。エラーは、クラッシュまたは動かなくなる状況を含んでよく、またはアプリケーションに関するフィードバックを提供する意図を手動で示すデバイスのユーザに起因することがある。そのようなイベントは、装置上で実行するオペレーティングシステムの構成要素によって等、イベントに関する診断情報を装置上に収集させうる。

30

【 0 0 3 3 】

動作 1 2 8 では、エラーに関する報告は、携帯デバイス 1 2 8 上に収集され、それは、携帯デバイスの動作およびデバイス上の各種アプリケーションを管理したオペレーティングシステムの制御下で生じうる行為である。例えば、オペレーティングシステムは、プロセスがデバイス上で終了するのに失敗した時を認識するためにプログラムされてもよく、そのようなイベントは、オペレーティングシステムによって動作を収集およびフォーマットする複数のデータをトリガしうる。例えば、トレース、スタック、レジスタ、または他のエンティティの現在のコンテンツは、エラーが生じた時にデバイス上で実行していた全ソフトウェア構成要素に関する識別子として判断されてもよい。そのような情報は次いで、合意標準 (agree-upon standard) に従って、情報の所定パケットへ共に収集されてもよい。

40

【 0 0 3 4 】

動作 1 3 0 では、報告または提供は、少なくとも概念的に、そのような報告に関して追跡および報告のタスクを行う中央サーバシステム上で存在するかのように示される。この例では、報告は、アプリケーションに対する特定のバグに対応するとして分類されており、そこでは、3 つの異なるバグが特定されている。そのような分類は例えば、関連するエラーが生じた時にどの部分のコードが実行していたかを特定して、そのコード部分に関する識別子を提供することによって生じることがあり、それにより特定の報告は、バグ報告の生成で共に加えられうる。特定のエラー提供 (error submission) に割り当てられる

50

べきバグの特定は、バグトレースシステムのユーザの手動分類によることを含む、他の方法でも実行できる。ここで示す通り、2つの他のバグは、特定されており、それらのバグに関して、それぞれ2および3つのエラー提案がある。故に、この例では、長期にわたり、初期アプリケーションは、8つの異なるエラー提案に対応するとして特定されている。

【0035】

そのような情報は、アプリケーションの開発者が知るのに重要なことがある。例えば、3つのバグのうち1つは、アプリケーションのユーザの満足度に重要なことがある。開発者がそのようなバグの種類を迅速に確認できた場合、開発者は、それを迅速に訂正して、コードに関する“パッチ”を供給できる（パッチは、アプリケーションをダウンロードした全ユーザへ自動的に押し出されるが、ユーザには、そのような更新を拒絶するための機会が与えられうる）。

10

【0036】

動作132および134では、そのような開発者は、全てのその開発者のバグ提供に関する報告を見ることを選択している。この例では、3つのバグが全システムで追跡されているが、動作134に示す通り、バグのうち2つは、動作120からのアプリケーションに関連付けられた。携帯アプリケーションマーケットまたは類似のサブシステムは、この例では故に、特定のエラーに関する情報と、システムによって追跡されているバグの種類を開発者に特定する情報とを示す報告を容易に生成しうる。

【0037】

特定のインスタンスでは、最終報告で提供されるデータは、携帯デバイスからの提供でアップロードされるデータとは実質的に異なることがある。例えば、個人的に識別可能な情報は、アップロードされたデータから引き離されてもよく、それにより開発者は、デバイスのユーザに関して特定の判断を行えない。一例では、スタックトレースおよびクラッシュ時のデバイスに関する類似の情報は、提供されるが、他の情報（例えば、問題が生じた時にユーザが操作していたデータを反映しうる情報）が渡されないことがある。また、詳細の特定レベルは、中央システムに関連するが、ほとんどの開発者に無関係であり、故に動作134で報告から引き離されてもよい。

20

【0038】

図2は、それらデバイス上のソフトウェアアプリケーションの動作に関するコンピューティングデバイスからの提供を追跡するためのシステム200の概略図である。システム200は、図1Aおよび1Bに関して上記で検討したそれらのように行為を実行するために使用されうる。システム200は、かなり単純な形式で明確性のために示され、特定の物理的実装では、かなり多数の追加の構成要素を含みうる。

30

【0039】

図を参照すると、システム200が示され、例えばラップトップおよびデスクトップコンピュータを使用しているとして示される各種開発者212は、アプリケーションサーバシステム202にアプリケーションを提供でき、見返りにお金と、その特定の各提供されたアプリケーションの動作に関するフィードバックとを受け取る。そのようなアプリケーションは、ここではアプリケーションをダウンロードおよび実行可能なワイヤレス携帯デバイスを介して示される、各種ユーザ210によって購入または取得されうる。

40

【0040】

開発者212およびユーザ210（またはより技術的には、そのデバイス）は一般に、直接互いに通信しないが、代わりにインターネット等のネットワーク204を使用して、アプリケーションサーバシステム202を介して通信しうる。そのようなオープン通信（open communication）により、広範囲の各種開発者タイプおよびユーザは、システム200を使用できるようになる。

【0041】

アプリケーションサーバシステム202の特定の構成要素は、システム200によってソフトウェアアプリケーションの使用を配置および追跡する機能を提供することができる。例えば、マーケットプレースフロントエンド214は、ユーザ210に利用可能なアプ

50

リケーションを表示するユーザインタフェースを生成でき、それによりユーザは、そのようなアプリケーションをレビュー、試行 (try)、およびダウンロード可能になる。マーケットプレースマネージャ 218 は、購入を追跡し、ユーザ 210 への請求および開発者 212 への送金額を生成し、およびアプリケーションストア実装の他の周知の機能により等、マーケットプレースフロントエンド 214 に対してバックエンドサポートを提供することができる。

【0042】

開発者フロントエンド 216 は、上記および下記で説明する通り、開発者がシステム 200 に登録し、アプリケーションをアップロードし、自分のアプリケーションを管理し (例えば、アプリケーションをアップデート、追加、削除し、自分のアプリケーションの売上額を確認し)、自分のアプリケーションの動作上のフィードバックを受信できるようにするユーザインタフェースを生成できる。そのようなフィードバックを提供するにおいて、フロントエンド 216 は、(例えば、クラッシュ、フリーズ、および手動のユーザ報告によって特定されるような) バグが生じた携帯デバイスからのデータのアップロードを管理しうるバグ報告生成装置 220 にアクセスできる。例えば、バグ報告生成装置は、ユーザデバイスからバグ報告データを受信する機能を持つことができ、バグレポジトリ 222 のようなストレージに適するようにそれをフォーマットし、次いで開発者が自分の特定のアプリケーションに関する報告確認を要求する時、格納されたデータの全部または一部を読み出す。

【0043】

バグ報告生成装置 220 または開発者フロントエンド 216、またはその2つの組み合わせは、開発者への配信に適するようにそのようなデータをフィルタ処理し、およびフォーマットしうる。例えば、2つの構成要素は、ウェブページを生成するために協働し、報告に対する開発者の要求に応じて開発者にページを送信できる。同様に、データは、データの配給されたフィードを要求した開発者に効率的にストリームされうるので、開発者は、自分がシステム 200 に到達するとリアルタイムでバグ報告を確認できる。そのようなリアルタイム報告は、開発者へのSMSメッセージ、XML転送 (それにより例えば、開発者は、独自のシステムで分析ツールを使用して、各種バグ報告を整理できる)、および他のそのようなメカニズム等、各種方法で生じることができる。

【0044】

各種データストアはまた、システム 200 によって格納、アクセス、および管理されるデータの例示的形式で図示される。例えば、上記の通り、バグレポジトリは、ユーザ 210 から生じたバグ報告に関する情報を格納できる。バグ情報は、バグイベントが生じた時に何が問題だったかを開発者または他の技術者が (少なくとも一部、少なくとも時々、または他のバグ報告と組み合わせて) 推測できるようにする診断情報を含みうる。バグレポジトリはまた、特定の環境では、開発者 212 ではなくシステム 200 の内部にアクセス可能であるべき情報を含みうる。そのような情報は、例えばバグ報告生成装置 220 および/または開発者フロントエンド 216 によって、適切な方法でフィルタ処理されうる。

【0045】

認証データベース 224 は、開発者 212 およびユーザ 210 に関する認証情報を格納できる。認証により、各種人々は、自分のアカウントに関連付けられた関連情報にアクセスし、システム 200 を介してアプリケーションの売買においてお金を支払い、または受け取ることができるようになる。また、認証は、特定のデータに対する特定のユーザによるアクセスの適切なレベルを判断するために使用されうる。例えば、第三者の開発者は、制限されたアクセスレベルに関連付けられた自分の認証を有することがあるので、自分のアプリケーションに対するバグの報告を要求する時、個人情報、アプリケーションのユーザに対して匿名性を維持するためにフィルタ処理される。これに対し、内部のユーザは、ユーザデバイス上の欠陥に関するデータへ比較的アクセスしやすい。

【0046】

最後に、アプリケーションデータストア 226 は、開発者 212 によってアップロードされたアプリケーション、およびシステム 200 の経営者によって初期にマーケットプレイス上に置かれたアプリケーションを含む、ユーザ 210 に利用可能に作成されるアプリケーションを格納する。

【0047】

サーバシステム 202 の外側では、ローカル開発者 206 がこの例でさらに示され、アプリケーションサーバシステム 202 と同じドメイン内でもよく、アプリケーションサーバシステム 202 に対してプライベートネットワーク 208 上で通信しうる。ローカル開発者は、アプリケーションサーバシステムを動作させる組織の従業員等、アプリケーションサーバシステム 202 の経営者によって信用されているプログラマでもよい。そのようなユーザは、各種方法でアプリケーションサーバシステム 202 によって特定されてもよく、開発者フロントエンド 216 は故に、開発者 206 がエラーデータへ比較的アクセスしやすくし、データを分析するためのより強固なツールへアクセスできる。即ち、開発者 206 は、開発者としてシステム 200 によって扱われるが、他の開発者 212 とは異なって扱われる。

【0048】

一例として、ユーザ 210 の一部は、アプリケーションサーバシステム 202 を提供した組織に提供される特定の個人識別可能なデータを有することに同意しているが、そのような情報を第三者と共有することを拒否している。ローカル開発者 206 は、開発者 212 よりもそのような状況で多くの情報にアクセスできる。そのような状況は、ユーザ 210 の視点から特に好ましいことがあり、その理由は、しばしばローカル開発者 206 は、システムで最も役立ち、普遍的に配置されているアプリケーションのいくつか（例えば、ワード処理および電子メール）を生成し、ユーザは、自分がプログラムを改善する必要がある全ての事項をそのような開発者に与えたいからである。

【0049】

図 3 は、ソフトウェア問題報告を管理するためのプロセスのフローチャートである。一般に、プロセスは、アプリケーションの開発者から各種コンピュータアプリケーションに関するコンピュータコードを受信し、アプリケーションマーケットプレイスの一般のメンバーによってダウンロードおよび将来的に購入させるために利用可能なアプリケーションを作成することを含む。各種アプリケーションがシステムの登録ユーザに関する各種コンピュータ上にインストールされた後、アプリケーションは、（例えば、クラッシュまたは他の問題がある時に）警告を生成し始め、そのような警告の情報は、アプリケーションマーケットプレイスの経営者に中継して返されることができ、情報は、イベントを有した特定のアプリケーションを提供した対応する開発者に利用可能に作成できるので、その開発者は、自分のアプリケーションに関するコードを改善および編集できる。

【0050】

プロセスは、ボックス 302 で開始し、プロセスは、開発者によるアプリケーションのアップロードを受信する。開発者は例えば、大工のための計算器、特定の専門職に対する検索ツール、または携帯コンピューティングデバイス上で実行できる他の類似のアプリケーション等、特定分野の特定のニッチ問題（niche problem）を解決するために単一のアプリケーションを書いた、自分の家で仕事をする個人のプログラマでもよい。開発者は代わりに、アプリケーションマーケットプレイス上で販売されるようなアプリケーションの開発を事業とする小さなソフトウェア会社のような、個人のグループを含んでもよい。一般に、開発者は、アプリケーションの販売によって生まれる収入、または各種ユーザにアプリケーションを分散することからもたらされる利益に対して正当な権利を主張する団体である。

【0051】

ボックス 304 では、アプリケーションは、マーケットプレイス上で利用可能に作成される。そのような動作は、類似の方法で生じることができ、アプリケーションのアイコンおよびテキスト記述のポスティングを含んでもよく、その両方は、開発者によってマーケ

10

20

30

40

50

ットプレースに提供される。アプリケーションは、支払いにより購入できるように作成され、または周知技術のフリーダウンロードができるように作成されうる。

【0052】

ボックス306で、アプリケーションのインスタンスは、アプリケーションのコピーを受信する権利（interest）を通信する購入者に配信される。結果として、アプリケーションのコピーは、アプリケーションストアまたはマーケットプレースにアクセスしているユーザのユーザアカウントに関連付けられた携帯デバイスにダウンロードされうる。ユーザは次いで、自分がアプリケーションを購入したのと同時に、またはその後か、しばらく経過後、アプリケーションをインストールし、自分のデバイス上でそれを実行するために、適切な行為をとりうる。故に、特定のアプリケーションをダウンロードしたデバイスのユーザ、およびアプリケーションをダウンロードした他のユーザは、アプリケーションにアクセスし、自由にアプリケーションと対話しうる。

10

【0053】

アプリケーションのそのような対話は、アプリケーションの動作に対して問題のある結果をもたらすことがある。1つのそのような結果は、クラッシュをもたらすことがあるアプリケーションの故障、またはアプリケーションのハングアップ（またはアプリケーションが実行しているプロセス）を含む。もう1つの結果は、ユーザによるアプリケーションに対する不満であり、その理由は、アプリケーションが不正確に実行されているからであり、またはアプリケーションが特定の改善で良好に実行するとユーザが思うからである。そのような状況では、ユーザの特定のクライアントデバイスは、デバイスから自動的に、またはユーザから手動でデータを収集し、イベントのパラメータを特徴づけることができる。上記説明した各種データは、オペレーティングシステムの構成要素またはアプリケーションによって特定の実装で収集でき、イベント時におけるデバイスの現在のスクリーンショットのような他のデータに追加して、上記検討したような情報と、ユーザがアプリケーションに追加したい一部の機能を示すワードのような、デバイスのユーザによってデバイスに入力されうる情報とを含むことができる。1度そのような診断情報が収集されると、それは、クライアントデバイスからサーバシステムに送信でき、サーバシステムは次いで、警告および他のクライアントデバイスから受信した他の警告に関する情報をフォーマットおよび保存できる（ボックス310）。

20

【0054】

そのようなフォーマットおよびフィルタ処理の実行では、中央サーバシステムはまた、アプリケーションに対して生じる同じバグを示すイベントを特定するために、異なるイベントのデータ間で共通のパラメータを特定できる。例えば、2つの異なるイベントに対するトレース情報が互いにかなり近く一致する場合、システムは、両方のイベントが共通のバグの結果だったと推測できる。2つのイベントが同じバグから生じたかどうかを判断するための特定のパラメータは、重み付けされてもよく、その一方で他のパラメータは、イベントが分類されるバグが特定のパラメータに関連するかどうかを判断するものでもよい。そのように特定されるバグの各々には、識別番号が割り当てられてもよく、その番号は、共通のバグを有すると判断される異なる警告データ間の相関関係を生成し、システムのソフトウェアの開発者に対して警告およびバグを記録するグループを提供するために後で

30

40

【0055】

ボックス312では、プロセスは、自分の対応するアプリケーションまたは複数のアプリケーションに関するバグ情報を、特定のアプリケーションまたは複数のアプリケーションの開発者に対してフォーマットおよび配信する。上記検討した通り、報告は、ウェブページの印刷報告、XML形式のようなデータのブロック、またはイベント若しくはイベントのグループがアプリケーションを動作している各種クライアントデバイスによってプロセスに報告される毎にデータ送信することを含むデータのストリームを含む、各種形式を取ることができる。このように、開発者は、ユーザのデバイス上に生じるバグ生成イベントに関する情報を取得でき、およびユーザおよび/またはユーザのデバイスがその情報を提

50

供できる。

【0056】

図4は、多数の開発者からのソフトウェアに関する問題の提供を管理するためのプロセスのスイムレーン図であり、ソフトウェアは、多数の携帯コンピューティングデバイスにわたってインストールされる。このプロセスは、図3のプロセスに類似するが、システムの特定の構成要素によって実行される特定の例示的動作に関して詳述される。プロセスは、ボックス402で開始し、開発者は、ソフトウェアアプリケーションを開発する。そのような開発は、アプリケーションのアーキテクチャ、アプリケーションに対するコードのドラフト、およびアプリケーションに対するコードの一部をパッケージおよび配列することを含みうる。一度アプリケーションが直ちに使用できるようになると、開発者は、類似の方法でアプリケーション格納サーバシステムに登録およびログインし（ボックス404）、サーバシステムは、この方法で開発者に対する対話型セッションを開始しうる（ボックス406）。アプリケーションのタイトル、アプリケーションに課せられる価格、アプリケーションのテキスト記述、ユーザのデバイスおよびアプリケーションストア上で提供されるべきアプリケーションのアイコン、および他の適切なデータ等、アプリケーションに関するメタデータの適切な形式を提供した後、ユーザまたは開発者は、必要なメタデータで自分のアプリケーションまたは複数のアプリケーションをアップロードしうる。

10

【0057】

ボックス410では、アプリケーションストアサーバは、アプリケーションを登録し、アプリケーションのパラメータを記述するデータとともにそれを格納するためのスペースを生成し、アプリケーションストア上で利用可能な方法でアプリケーションをホストする。

20

【0058】

しばらく経ったある時点で、アプリケーションストアを閲覧しているユーザは、アプリケーションを発見し、それを購入したいと思う。ボックス414では、ユーザは、アプリケーションストアに対して登録およびログインし、その後ユーザとのセッションを開始する（ボックス416）。

【0059】

ボックス418では、ユーザは、アプリケーションのアイコン上をクリックする等、アプリケーションをダウンロードするコマンドを供給し、次いでユーザがアプリケーションを購入および取得したいことを確認するために特定のスクリーン上の選択可能な制御を選択する。ボックス420では、アプリケーションストアは、ユーザの認証をチェックし、ユーザがアプリケーションストアに対するアカウントを有することを確かめ、次いでユーザがそのようなアカウントを有する場合、ユーザにアプリケーションを提供する。代わりに、サーバは、ユーザがアカウントを有しないと判断することがあり、将来ダウンロードしたアプリケーションに対して支払うユーザからクレジットカード情報を要求することを含む、従来の方法でユーザからのそのような情報を要求しうる。

30

【0060】

一度ユーザがアプリケーションをダウンロードすると、ユーザがその使用を開始し、そのような使用は、特定の環境では、ボックス422に示す通り、エラーまたは問題となるイベントを生成することがある。エラーイベントの例は、上記にて提供され、デバイスのフリーズ、デバイスのクラッシュ、またはアプリケーションに関して提供された改善または問題に関するユーザからの手動の報告を含んでもよい。ユーザのデバイスは次いで、ボックス424でバグサーバに対して、イベントに関する診断情報と共にそのようなイベントを報告し、データは、受信および記録されうる。クライアントデバイスからのそのような送信は、イベントを被る特定のアプリケーションによって、またはデバイス上に常駐する異なるアプリケーションによって、オペレーティングシステム構成要素を介して生じうる。デバイスおよびバグサーバ間の特定の対話は、API標準によって確立されてもよく、データが適切に送信されるような標準に従ってもよい。

40

【0061】

50

プロセスは次いで、しばらく待機してもよく、他のユーザは、自分のデバイス上でアプリケーションを実行してもよく、その後それらデバイス上の問題に関するデータを報告してもよい。しかし、しばらく経ったある時点で、開発者は、アプリケーションが適切に開いているかを確認したいと望むことがあり、故に最後にログインし（ボックス 2 2 6）、この時でも、開発者は、バグサーバからの情報を備えることができ、ボックス 4 2 8 でセッションを開始しうる。1 つの共通の対話は、開発者によって動作されるコンピューティングデバイスによって作成される 1 つまたは複数の報告の要求（ボックス 4 3 0）を含んでもよく、バグサーバはボックス 4 3 2 で、要求された報告を開発者に戻す機能があってもよい。そのような機能は、バグサーバによって特定された特定のポストに従ってエラーイベントデータを整理すること、および開発者が知らされた判断を作成するために必要なデータのみを受信し、アクセスする必要のないデータを受信しないように、データをフィルタ処理することを含む。最後に、ボックス 4 3 4 では、開発者のデバイスは、ウェブページ等の上に記録または報告を表示し、それにより開発者は、1 つまたは複数のデバイス上で生じうるエラーに対処するために、それらをレビューでき、自分のソフトウェアを更新するか否かおよびその方法に関する判断を行うことができる。

10

【 0 0 6 2 】

図 5 A および 5 B は、ソフトウェアアプリケーションに関して動作上の問題を受けた携帯コンピューティングデバイスの例示的なスクリーンショットである。一般に、これらのスクリーンショットは、エラーイベントが生じた時、オペレーティングシステム構成要素によって、またはアプリケーションによって等、クライアントデバイス上で生成されうるスクリーンの 2 つの例を示す。図 5 A のスクリーンショットは、特定のプロセスが予想外に停止したことによって、生成されている警告スクリーンを示す。スクリーンショットは、何が発生したかについてユーザに示し、次いで対応するための 2 つのオプションをユーザに与える。特に、“強制終了 (force close)” を選択して、ユーザは、プロセスをシャットダウンするか、ひいてはユーザがデータを失うかもしれないことを理解した上でアプリケーションをシャットダウンできる。ユーザはまた、問題に関する情報を報告することを選択でき、ユーザは、偏見がなく、アプリケーションの開発者を援助したいと思うか、またはオペレーティングシステムがソフトウェアを改善することで、類似の予想外の停止が将来発生しないようにする。

20

【 0 0 6 3 】

ユーザが“報告”オプションを選択すると、図 5 B のスクリーンショットを表示させる。ここで、ユーザは、どのように問題が中央サーバシステムに報告されるのかを制御するパラメータが示される。特に、送信に適用される適切な個人情報保護 (privacy policy) は、ユーザレビューのために表示されることがあり、ユーザは、個人情報保護を訂正または更新するために適切な制御を行うことができる。また、ユーザには、空テキストボックスが提示され、問題のコンテキストをオプションとして記述でき、例えば、“私は、abc アプリケーションに xyz をタイプし、一方同時に電話で話をして、Handbrake でビデオを変換し、SETI に関してナンバークランピングを行い、フルレス (full res) 一人称シューティングゲームをプレイしていて、これといった理由もなく、アプリケーションが私の前で停止した。”そのようなテキストは、問題に関する他の診断情報を備えるフィールドとしてやり過ごされてもよい。特定の環境では、ユーザは、バグが特に処理しにくく、開発者がユーザへの電子メールまたは電話を必要とする場合、自分の連絡先を開発者に提供する機会が与えられてもよい。代わりに、ユーザは、初めから任意の連絡先を提供するように尋ねられないが、開発者は、コメントをレビューする時、連絡先に関する要求を置くことができ、システムは次いで、(thew) ユーザが応答しうるメッセージを生成する。

30

40

【 0 0 6 4 】

ユーザは次いで、問題に関してアップロードされる全データのビューが与えられる“レビュー”ボタン、または情報のそのようなアップロードを引き起こす“送信”を選択できる。

【 0 0 6 5 】

50

図 6 A ~ 6 C は、ソフトウェア追跡システムによって生成される開発者報告の例示的なスクリーンショットである。図 6 A は、application com.android.launcher2のバグに関する要約統計である。例えば、クライアントデバイスからの報告または提供の全数が、フリーズおよびクラッシュの両方に対する提供の現在比率として示される。

【 0 0 6 6 】

もう 1 つのインスタンスでは、デバイスのバッテリー電力および使用量に関する情報は、実行アプリケーションに関する情報と共に報告されうる。例えば、バッテリーの電力が突然下がる場合、診断情報は、デバイスからアップロードできるので、中央の研究者 (central researcher) は、選挙需要 (electoral demand) の原因を判断できる。例えば、特定のアプリケーションは、そのようなイベントが生じた時はいつでもアクティブであるとしてしばしば示される場合、中央システムは、アプリケーションがバッテリー消耗の原因であると推測しうる。

10

【 0 0 6 7 】

図 6 B は、複数の異なるアプリケーションまたはコードファイルに関するが、類似する要約報告である。例えば、コード名は、それが失敗した時にそのコンテキストと、そのような失敗に関する報告数と、そのようなイベントの現在比率と共に示される。

【 0 0 6 8 】

図 6 C は、特定のアプリケーションの特定の失敗に関する詳細報告を示し、そして失敗のコンテキストに関するさらなる情報を示す。例えば、デバイスに関するスタックトレースは、特定の他の有益な情報として報告で示される。情報の一部は、ハイパーリンクされてもよく、それによりユーザは、何がここで表示されているかに関する追加のソースデータを確認できる。

20

【 0 0 6 9 】

図 7 は、包括的なコンピュータデバイス 7 0 0 および包括的な携帯コンピュータデバイス 7 5 0 の例を示し、ここで説明される技術に使用されうる。コンピューティングデバイス 7 0 0 は、ラップトップ、デスクトップ、ワークステーション、個人デジタル補助装置、サーバ、ブレードサーバ、メインフレーム、および他の適切なコンピュータ等、デジタルコンピュータの各種形式を示すことを意図する。コンピューティングデバイス 7 5 0 は、個人デジタル補助装置、セルラー電話、スマートフォン、および他の類似のコンピューティングデバイス等、携帯デバイスの各種形式を示すことを意図する。ここに示す構成要素、その接続および関係、およびその機能は、単に例示目的であって、本明細書で説明および/または請求される発明の実施形態を限定することを目的としない。

30

【 0 0 7 0 】

コンピューティングデバイス 7 0 0 は、プロセッサ 7 0 2、メモリ 7 0 4、格納デバイス 7 0 6、メモリ 7 0 4 と高速拡張ポート 7 1 0 とに接続する高速インタフェース 7 0 8、および低速バス 7 1 4 と格納デバイス 7 0 6 とに接続する低速インタフェース 7 1 2 を含む。構成要素 7 0 2、7 0 4、7 0 6、7 0 8、7 1 0、および 7 1 2 の各々は、各種バスを使用して相互接続され、共通のマザーボード上または他の適切な方法で搭載される。プロセッサ 7 0 2 は、メモリ 7 0 4 内または格納デバイス 7 0 6 上に格納される命令を含む、コンピューティングデバイス 7 0 0 内で実行に関する命令を処理し、高速インタフェース 7 0 8 に連結されるディスプレイ 7 1 6 のような外部入/出力デバイス上に GUI に関するグラフィカル情報を表示することができる。他の実施形態では、多数のメモリおよびメモリの種類と共に、多数のプロセッサおよび/または多数のバスを適切に使用できる。また、多数のコンピューティングデバイス 7 0 0 は、必要な動作の一部を提供する各デバイス (例えば、サーババンク、ブレードサーバのグループ、またはマルチプロセッサシステム) に接続されてもよい。

40

【 0 0 7 1 】

メモリ 7 0 4 はコンピューティングデバイス 7 0 0 内に情報を格納する。1 つの実施形態では、メモリ 7 0 4 は、揮発性メモリユニットまたは複数のユニットである。もう 1 つの実施形態では、メモリ 7 0 4 は、不揮発性メモリユニットまたは複数のユニットである

50

。メモリ704はまた、磁気または光学ディスク等、コンピュータ読取可能な媒体の他の形式でもよい。

【0072】

格納デバイス706は、コンピューティングデバイス700に大容量ストレージを提供できる。1つの実施形態では、格納デバイス706は、フロッピー（登録商標）ディスクデバイス、ハードディスクデバイス、光学ディスクデバイス、またはテープデバイス、フラッシュメモリまたは他の類似の個体メモリデバイス、またはストレージエリアネットワークまたは他の構成のデバイスを含むデバイスアレイ等、コンピュータ読取可能な媒体であるか、または含んでもよい。コンピュータプログラム製品はまた、情報キャリアに有形に具現されうる。コンピュータプログラム製品はまた、上記説明したような1つまたは複数10の方法を実行時に行う命令を含みうる。情報キャリアは、メモリ704、格納デバイス706、プロセッサ702上のメモリ、または伝搬信号等、コンピュータまたは機械読取可能な媒体である。

【0073】

高速コントローラ708は、コンピューティングデバイス700に対する帯域幅集中動作（bandwidth-intensive operation）を管理し、一方低速コントローラ712は、比較的低い帯域幅集中動作を管理する。機能のそのような割り当ては、単なる例示である。1つの実施形態では、高速コントローラ708は、メモリ704、（グラフィックプロセッサまたはアクセラレータを介して）ディスプレイ716、および高速拡張ポート710に20連結され、ポート710は、各種拡張カード（図示せず）を受け入れることができる。実施形態では、低速コントローラ712は、格納デバイス706および低速拡張ポート714に連結される。各種通信ポート（例えば、USB、ブルートゥース、イーサネット（登録商標）、ワイヤレスイーサネット（登録商標））を含みうる低速拡張ポートは、キーボード、ポインティングデバイス、スキャナ、または例えばネットワークアダプタを介するスイッチまたはルータのようなネットワークデバイス等、1つまたは複数の入/出力デバイスに連結できる。

【0074】

コンピューティングデバイス700は、図示の通り、複数の異なる形式で実装されうる。例えば、それは、標準サーバ720として、またはそのようなサーバのグループで複数30回実装されてもよい。それはまた、ラックサーバシステム724の一部として実装されてもよい。また、それは、ラップトップコンピュータ722のような個人コンピュータで実装されてもよい。代わりに、コンピューティングデバイス700からの構成要素は、デバイス750のような携帯デバイスの他の構成要素（図示せず）に結合されてもよい。そのようなデバイスの各々は、1つまたは複数のコンピューティングデバイス700、750を含んでもよく、全システムは、互いに通信する多数のコンピューティングデバイス700、750から構成されてもよい。

【0075】

コンピューティングデバイス750は、他の構成要素の中で、プロセッサ752、メモリ764、ディスプレイ754のような入/出力デバイス、通信インタフェース766、およびトランシーバ768を含む。デバイス750はまた、追加のストレージを提供する40ためにマイクロドライブまたは他のドライブのような格納デバイスを提供してもよい。各構成要素750、752、764、754、766、および768は、各種バスを使用して相互接続され、いくつかの構成要素は、共通のマザーボード上、または他の適切な方法で搭載されうる。

【0076】

プロセッサ752は、メモリ764に格納される命令を含む命令を、コンピューティングデバイス750内で実行できる。プロセッサは、分離した多数のアナログおよびデジタルプロセッサを含むチップのチップセットとして実装されてもよい。プロセッサは例えば、ユーザインタフェース、デバイス750によって動作するアプリケーション、およびデバイス750によるワイヤレス通信の制御等、デバイス750他の構成要素の調整を提供50

しうる。

【0077】

プロセッサ752は、ディスプレイ754に連結される制御インタフェース758およびディスプレイインタフェース756を介してユーザと通信しうる。ディスプレイ754は例えば、TFT LCD（薄膜トランジスタ液晶ディスプレイ）またはOLED（有機光ダイオード）ディスプレイ、または他の適切なディスプレイ技術でもよい。ディスプレイインタフェース756は、グラフィカルおよび他の情報をユーザに提示するために、ディスプレイを駆動する適切な回路を含みうる。制御インタフェース758は、ユーザからコマンドを受け取り、プロセッサ752へ提供するためにそれらを変換しうる。また、外部インタフェース762は、プロセッサ752と通信するよう提供され、他のデバイスに
10 対するデバイス750の近接通信を可能にする。外部インタフェース762は例えば、いくつかの実施形態でワイヤード通信を提供し、または他の実施形態でワイヤレス通信を提供してもよく、多数のインタフェースがさらに使用されてもよい。

【0078】

メモリ764は、コンピューティングデバイス750内に情報を格納する。メモリ764は、1つまたは複数のコンピュータ読取可能な媒体またはメディア、揮発性メモリユニットまたは複数のユニット、または不揮発性メモリユニットまたは複数のユニットとして実装されてもよい。拡張メモリ774がさらに提供され、拡張インタフェース772を介してデバイス750に接続でき、拡張インタフェース772は例えば、SIMM（シングルインラインメモリモジュール）カードインタフェースを含んでもよい。そのような拡張
20 メモリ774は、デバイス750に対して追加の格納スペースを提供してもよく、またはさらにデバイス750に関するアプリケーションまたは他の情報を格納してもよい。特に、拡張メモリ774は、上記説明したプロセスを実行または補足するための命令を含むことができ、セキュア情報も含みうる。故に例えば、拡張メモリ774は、デバイス750に関するセキュリティモジュールとして提供されてもよく、デバイス750の安全な使用を可能にする命令でプログラムされてもよい。また、セキュアアプリケーションは、ハッキング不可能な方法でSIMMカード上に識別情報を設定する等、追加情報と共にSIMMカードを介して提供されてもよい。

【0079】

メモリは例えば、下記の通り、フラッシュメモリおよび/またはNVRAMメモリを含んでもよい。1つの実施形態では、コンピュータプログラム製品は、情報キャリアに有形に具現される。コンピュータプログラム製品は実行時、上記説明したような1つまたは複数の方法を行う命令を含む。情報キャリアは、メモリ764、拡張メモリ774、プロセッサ752上のメモリ、例えばトランシーバ768または外部インタフェース762上で受信可能な伝搬信号等、コンピュータまたは機械読取可能な媒体である。
30

【0080】

デバイス750は、通信インタフェース766を介してワイヤレス通信でき、インタフェース766は、必要に応じてデジタル信号処理回路を含みうる。通信インタフェース766は、GSM（登録商標）音声呼、SMS、EMS、またはMMSメッセージング、CDMA、TDMA、PDC、WCDMA、CDMA2000またはGPRSその他等、
40 各種モードまたはプロトコル下の通信を提供できる。そのような通信は例えば、無線周波トランシーバ768を介して生じうる。また、短距離通信は、ブルートゥース、WiFi、または他のそのようなトランシーバ（図示せず）を使用する等で生じうる。また、GPS（全地球測位システム）受信モジュール770は、デバイス750に追加のナビゲーションおよび位置関連ワイヤレスデータを提供でき、デバイス750上で動作するアプリケーションによって適切に使用されうる。

【0081】

デバイス750は、音声コーデック760を使用して音声認識可能に通信でき、そのコーデックは、ユーザから話された情報を受信して有益なデジタル情報にそれを変換しうる。音声コーデック760は同様に、スピーカー、例えばデバイス750のハンドセット等
50

を介して、ユーザに対して音声認識可能な音を生成しうる。そのような音は、音声電話呼からの音を含んでもよく、記録された音（例えば、音声メッセージ、音楽ファイル等）を含んでもよく、またデバイス750上で動作するアプリケーションによって生成された音を含んでもよい。

【0082】

コンピューティングデバイス750は、図示の通り、複数の異なる形式で実装されてもよい。例えば、携帯電話780として実装されてもよい。また、スマートフォン782、個人デジタル補助装置、または他の類似の携帯デバイスの一部として実装されてもよい。

【0083】

ここに記載したシステムおよび技法の各種実装は、デジタル電子回路、集積回路、専用に設計されたASIC（特定用途向け集積回路）、コンピュータハードウェア、ファームウェア、ソフトウェア、および/またはその組み合わせで実現できる。これら各種実装は、少なくとも1つのプログラマブルプロセッサを含むプログラマブルシステム上で実行可能および/または解釈可能な1つまたは複数のコンピュータプログラムでの実装を含むことができ、それは、専用または汎用目的でもよく、格納システム、少なくとも1つの入力デバイス、および少なくとも1つの出力デバイスに対して、データおよび命令を受信し、データおよび命令を送信するように結合される。

【0084】

これらコンピュータプログラム（また、プログラム、ソフトウェア、ソフトウェアアプリケーションまたはコードとして知られる）は、プログラマブルプロセッサに対する機械命令を含み、高レベルプロシージャおよび/またはオブジェクト指向プログラム言語、および/またはアセンブリ/機械言語で実装されうる。ここで使用される通り、用語“機械読取可能な媒体”は、機械読取可能な信号として機械命令を受信する機械読取可能な媒体を含む、プログラマブルプロセッサに機械命令および/またはデータを提供するために使用される任意のコンピュータプログラム製品、装置、および/またはデバイス（例えば、磁気ディスク、光学ディスク、メモリ、プログラマブル論理デバイス（PLD））に言及する。用語“機械読取可能な信号”は、プログラマブルプロセッサに機械命令および/またはデータを提供するために使用される任意の信号に言及する。

【0085】

ユーザとの対話を提供するために、ここで説明されたシステムおよび技法は、ユーザに情報を表示するためのディスプレイデバイス（例えば、CRT（ブラウン管）またはLCD（液晶ディスプレイ）モニター）、およびユーザがコンピュータに入力を提供できるポインティングデバイス（例えば、マウスまたはトラックボール）を有するコンピュータ上で実装されうる。他の種類のデバイスを使用してユーザとの対話を提供してもよく、例えば、ユーザに提供されるフィードバックは、感覚フィードバックの任意形式（例えば、視覚フィードバック、音声フィードバック、または触覚フィードバック）でもよく、ユーザからの入力は、音響、音声、または触覚入力を含む任意形式で受信されうる。

【0086】

ここで説明されたシステムおよび技法は、コンピューティングシステムで実装でき、それは、バックエンド構成要素（例えば、データサーバ）を含み、ミドルウェア構成要素（例えば、アプリケーションサーバ）を含み、フロントエンド構成要素（例えば、ユーザがここで説明されたシステムおよび技法の実装と対話できるグラフィカルユーザインタフェースまたはウェブブラウザを有するクライアントコンピュータ）を含み、またはそのようなバックエンド、ミドルウェア、またはフロントエンド構成要素の任意の組み合わせを含む。システムの構成要素は、デジタルデータ通信の任意形式（例えば、通信ネットワーク）または媒体によって相互接続されてもよい。通信ネットワークの例は、ローカルエリアネットワーク（LAN）、広域ネットワーク（WAN）、およびインターネットを含む。

【0087】

コンピューティングシステムは、クライアントおよびサーバを含みうる。クライアントおよびサーバは一般に、互いに離れており、通信ネットワークを介して通常、対話する。

10

20

30

40

50

クライアントおよびサーバの関係は、各コンピュータ上で動作し、互いにクライアントサーバ関係を有するコンピュータプログラムによって生じる。

【0088】

複数の実施形態が説明された。一方で、各種修正が本発明の精神および範囲から逸脱しないで行えることが分かる。例えば、この明細書の大部分は、電話ダイアリングアプリケーションに関して説明されているが、英数字だけでなく、グラフィカルアイコンおよびマクロを含むキーパッド等、他の形式のアプリケーションおよびキーパッドレイアウトも適用できる。

【0089】

また、図に示す論理フローは、所望の結果を達成するために、図示されている特定の順番、即ち順次の順番を要しない。また、他のステップが提供されてもよく、即ちステップが、説明されたフローから削除されてもよく、他の構成要素が、説明されたシステムに対して追加または除去されてもよい。従って、他の実施形態は、以下の特許請求の範囲内にある。

【符号の説明】

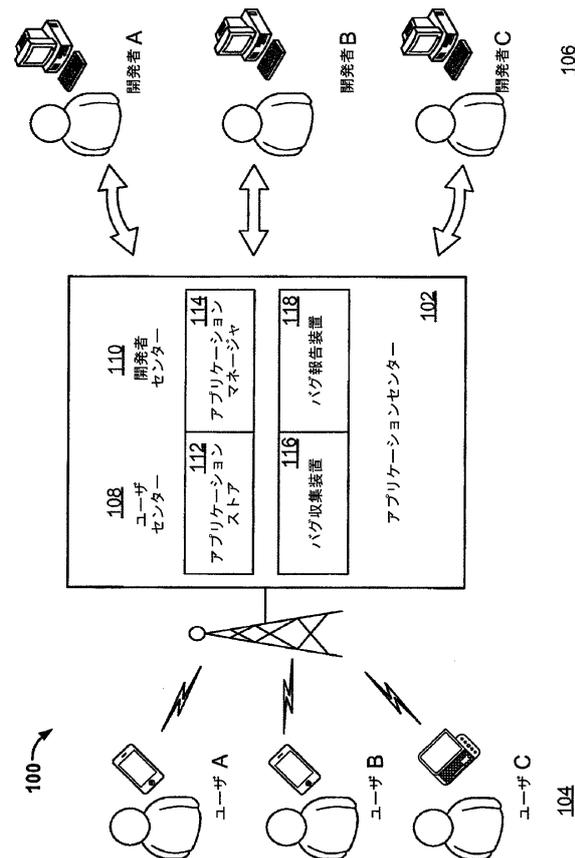
【0090】

- 102：アプリケーションセンター
- 108：ユーザセンター
- 110：開発者センター
- 112：アプリケーションストア
- 114：アプリケーションマネージャ
- 116：バグ収集装置
- 118：バグ報告装置

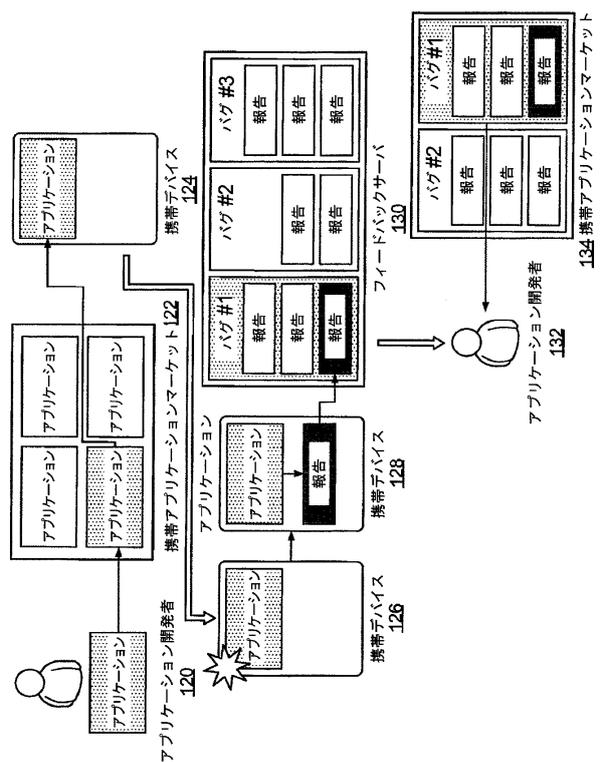
10

20

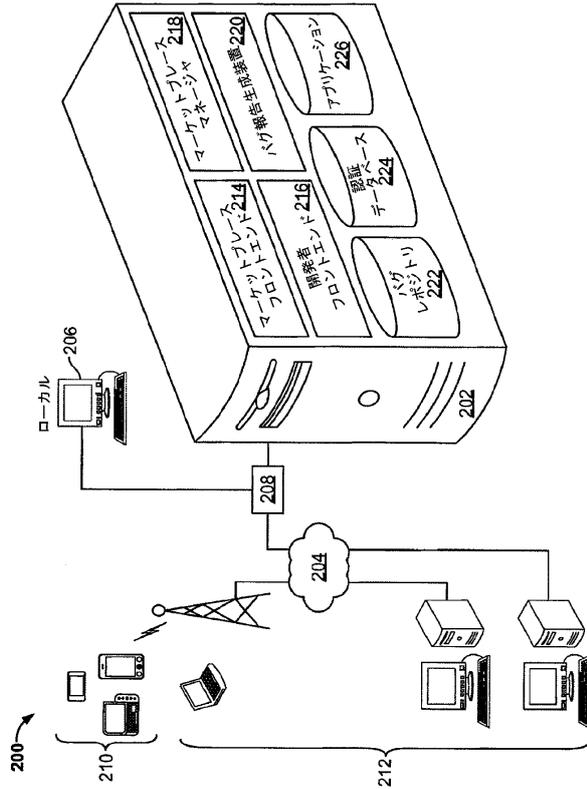
【図1A】



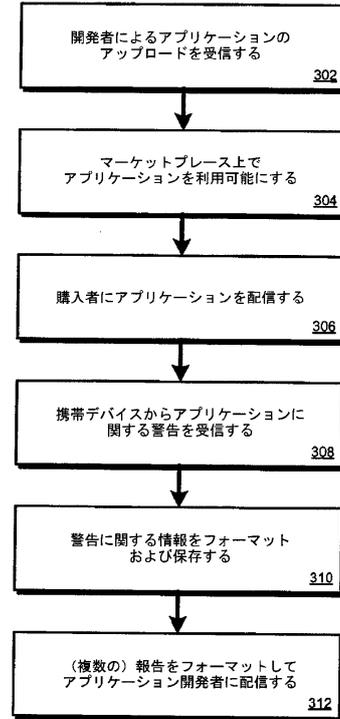
【図1B】



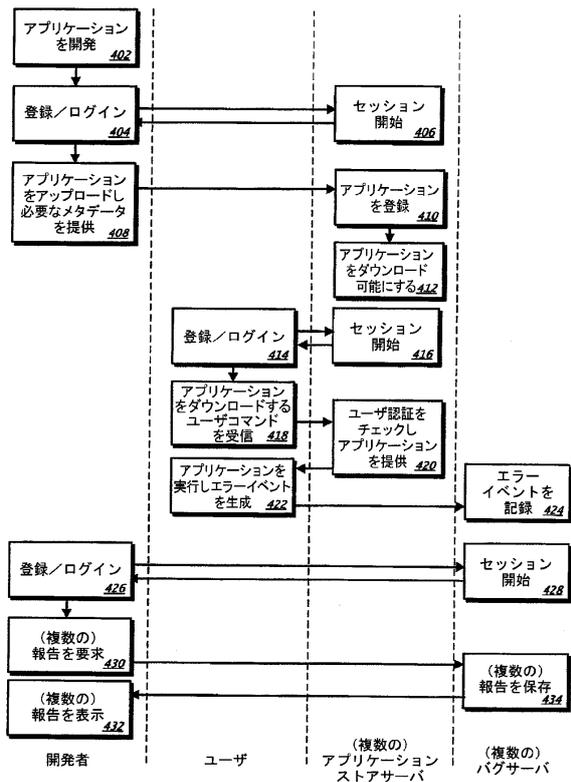
【図2】



【図3】



【図4】



【図5A】

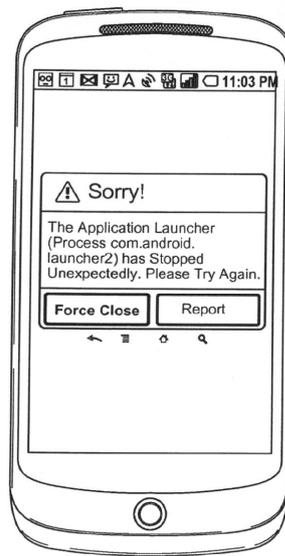


FIG. 5A

【 5 B 】

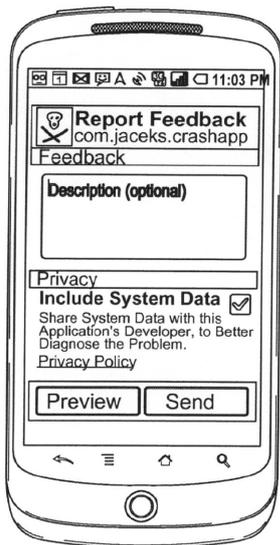


FIG. 5B

【 6 A 】

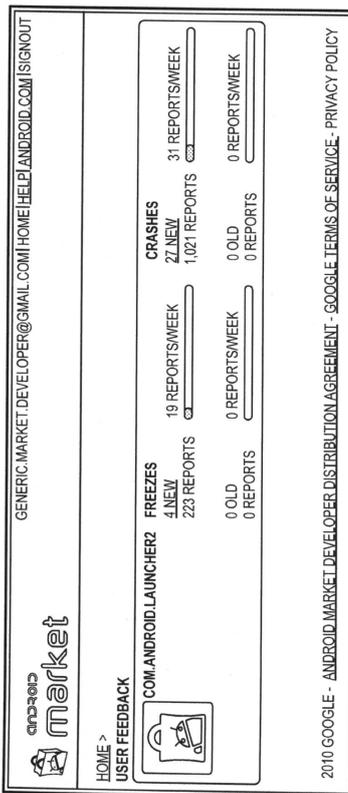


FIG. 6A

【 6 B 】

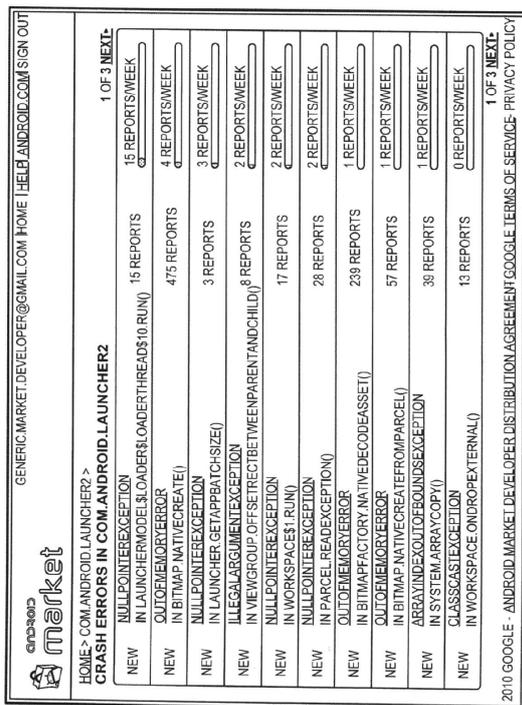


FIG. 6B

【 6 C 】

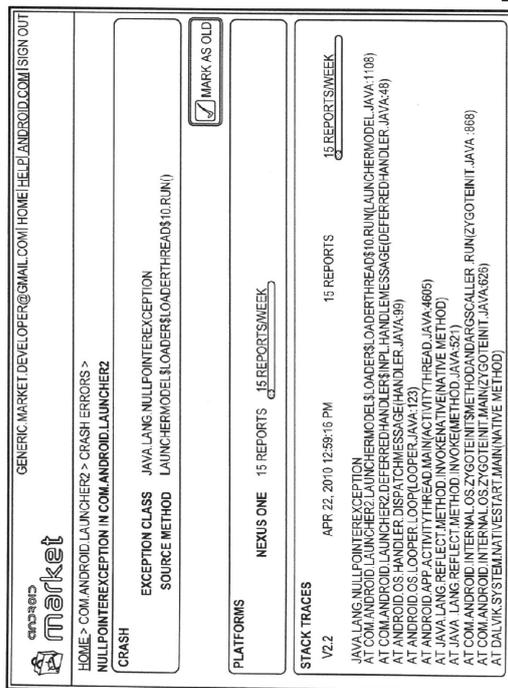


FIG. 6C

【 7 】

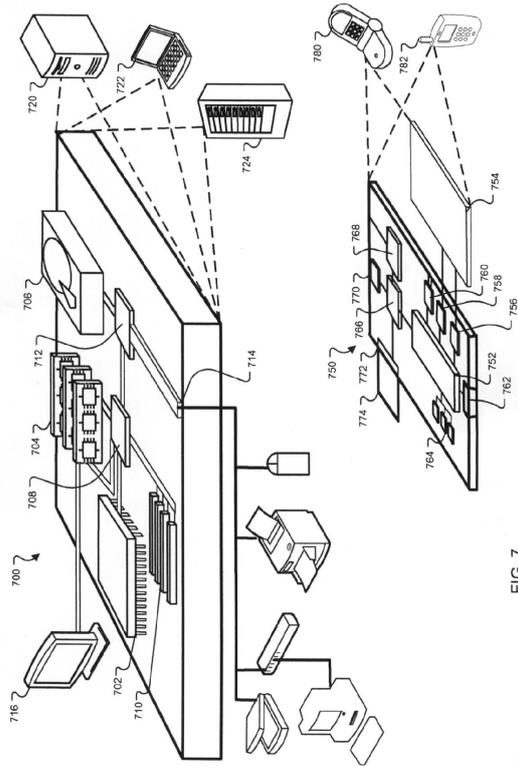


FIG. 7

フロントページの続き

(72)発明者 ダビド・ドゥダ

ポーランド・32-087・ジェロンキ・ウリツァ・クフィアトフ・ポルニフ・15ツェ

審査官 多賀 実

(56)参考文献 米国特許出願公開第2007/0168343(US, A1)

特開2007-272443(JP, A)

島影 正俊 外2名, 「ソフトウェアの協調開発コミュニティウェアの実装とその試行」, マルチメディア, 分散, 協調とモバイル(DICOMO)シンポジウム論文集 1997年~2006年版 Ver.1.1 [DVD-ROM], 社団法人情報処理学会, 2006年 8月31日, 第2002巻, 第9号, pp.425-428

有山 圭二, 「Android開発 A to Z 開発環境のセットアップからサンプルアプリ構築, そしてAndroid Marketへの公開まで!」, Software Design 2010年3月号 特別付録小冊子, (株)技術評論社, 2010年 3月18日, 通巻299号(発刊233号), pp.31-35

(58)調査した分野(Int.Cl., DB名)

G06F 9/44

G06F 11/36