



República Federativa do Brasil

Ministério do Desenvolvimento, Indústria,
Comércio e Serviços

Instituto Nacional da Propriedade Industrial

(11) BR 112015027087-5 B1

(22) Data do Depósito: 18/06/2013

(45) Data de Concessão: 18/04/2023

(54) Título: MÉTODO DE ALTERAÇÃO DE INTERFACE AUTOMATIZADA (VERSÕES)

(51) Int.Cl.: G06F 9/445; G06F 17/40.

(30) Prioridade Unionista: 26/04/2013 RU 2013119434.

(73) Titular(es): LIMITED LIABILITY COMPANY "1C".

(72) Inventor(es): DENIS ANATOLYEVICH PAVLENKO; ODAY HALIKOVICH DERUT.

(86) Pedido PCT: PCT RU2013000505 de 18/06/2013

(87) Publicação PCT: WO 2014/175767 de 30/10/2014

(85) Data do Início da Fase Nacional: 26/10/2015

(57) Resumo: MÉTODO DE ALTERAÇÃO DE INTERFACE AUTOMATIZADA (VERSÕES). Esta invenção relaciona-se com a tecnologia de informação, em especial com o método de alteração de interface automatizada. Ela melhora o conjunto de ferramentas tecnológicas e permite configurar a interface do usuário de forma flexível na fase de implementação e execução. Este resultado tecnológico é alcançado através do aplicativo do método de alteração de interface automatizada que compreende as seguintes fases: a criação de um elemento do sistema separado preliminarmente em fase de desenvolvimento, especificando seus elementos componentes dependentes e salvando tal elemento separado no dispositivo de armazenamento; formando um elemento especial para armazenar os valores do elemento do sistema separado; definindo o valor do referido elemento do sistema especial na fase de implementação ou execução do aplicativo; baixar esse elemento sistema separado na fase de execução do aplicativo; a obtenção do valor da elemento do sistema separado a partir do elemento de armazenamento de valor especial; reconfigurando elementos de interface do usuário relacionados com elementos dependentes componentes ao utilizar o valor obtido do elemento de sistema separado.

MÉTODO DE ALTERAÇÃO DE INTERFACE AUTOMATIZADA (VERSÕES).

Campo de aplicação da Invenção

[001] Esta invenção relaciona-se com a tecnologia de informação e, mais especificamente, com o método de alteração de interface automatizada.

Estado Da Técnica

[002] O processo de alteração de interface automatizada requer uma maneira flexível e amigável ao usuário para a configuração da interface do usuário na fase de implementação ou execução de aplicativos de negócios em computadores eletrônicos, como computadores pessoais, laptops etc.

[003] “Implementação” aqui refere-se a um processo de implantação do software em computadores eletrônicos e configuração de acordo com um ambiente específico, a fim de ser executado pelos usuários.

[004] Existem diferentes maneiras de lidar com dito desafio.

[005] O Pedido de Patente americano nº 2011252397 (publicação em 13 de outubro de 2011) oferece um método de alteração de interface automatizada que compreende receber um pedido de um usuário; obter todas as configurações de sistemas capazes de fornecer o serviço solicitado a partir de um modelo de recurso dinâmico; obter uma configuração correspondente; reconfigurar os recursos do sistema, com base na configuração obtida; e fornecer o serviço solicitado. Como resultado, o referido método faz com que seja possível fornecer serviços otimizados para um ambiente que varia em tempo real, sem a interferência do usuário.

[006] No entanto, este método é de uso limitado visto que a obtenção de todas as configurações de sistema para se escolher uma configuração correspondente e reconfigurar os recursos do sistema, com base na configuração obtida, prolonga o processo de alteração da interface do usuário e requer ainda recursos significativos para armazenar todas as configurações possíveis.

Breve descrição da Invenção

[007] Assim, há uma necessidade de se melhorar o conjunto de ferramentas tecnológicas através do desenvolvimento de um método rápido e universal de alteração de interface

automatizada que permitiria ao usuário configurá-lo por ele / ela mesmos de uma forma flexível e de fácil utilização quando da implementação ou da execução do aplicativo.

[008] Para alcançar este objetivo e o resultado técnico especificado, a primeira forma de realização da invenção oferece um método de alteração de interface automatizada que compreende as seguintes fases: criar um elemento de sistema separado preliminarmente na fase de desenvolvimento, especificar seus elementos de composição dependentes e salvar tais elementos separados no dispositivo de armazenamento; formar um elemento especial para armazenar os valores do elemento de sistema separado; definir o valor do referido elemento especial do sistema na fase de implementação ou execução do aplicativo; baixar o elemento de sistema separado na fase de execução do aplicativo; obter valor do elemento de sistema separado do elemento de armazenamento de valor especial; reconfigurar elementos de interface do usuário relacionados com elementos de composição dependentes ao utilizar o valor obtido do elemento de sistema separado.

[009] A seguinte forma de realização da invenção torna possível alcançar resultado técnico especificado mediante o aplicativo do método de alteração de interface automatizada que compreende as seguintes fases: criar um elemento de sistema separado preliminarmente em fase de desenvolvimento, especificar seus elementos de composição dependentes e salvar tal elemento separado no dispositivo de armazenamento; criar um elemento para parametrizar valores do elemento de sistema separado; formar um elemento destinado à armazenar combinações de valores de parâmetros com valores do elemento de sistema separado; entrar com as combinações de valores de parâmetros com os valores do elemento de sistema separado na fase de implementação ou execução do aplicativo; definir os valores dos parâmetros na fase de execução do aplicativo; selecionar o valor do elemento de sistema separado que corresponde ao valor do parâmetro específico do elemento de armazenamento de combinações de valores de parâmetros com os valores do elemento de sistema separado; reconfigurar elementos de interface do usuário relacionados com elementos de composição dependentes ao utilizar o valor obtido do elemento de sistema separado.

[010] O atributo específico de ambas as formas de realização é que, onde o elemento de sistema depende de vários elementos separados, partes da interface conectadas a esse elemento de sistema serão exibidas se for permitido por, pelo menos, um desses elementos separados.

Breve descrição dos desenhos

[011] Fig. 1 é um diagrama de blocos lógico que ilustra o processo de alteração de interface automatizada de acordo com a primeira versão da presente invenção.

[012] Fig. 2 é um diagrama de blocos lógico que ilustra o processo de alteração de interface automatizada de acordo com a segunda versão da presente invenção.

Descrição detalhada das formas de realização da Invenção

[013] Esta invenção pode ser implementada em computadores eletrônicos (por exemplo, computadores pessoais etc.) em um sistema de computação (a seguir mencionado como o “sistema”) projetado para automatização dos processos relacionados às atividades econômicas e organizacionais, vários tipos de procedimentos de contabilidade, análises financeiras e econômicas em uma empresa em qualquer área de negócio. O sistema existente como um conjunto de arquivos invariáveis compreende uma combinação de várias ferramentas de software aplicadas tanto por desenvolvedores e usuários. O sistema está logicamente dividido em duas partes que interagem uma com a outra. Eles são uma configuração, isto é, a capacidade de configurar o sistema de acordo com o ambiente de qualquer empresa e tarefas específicas a serem realizadas, e uma plataforma para gerenciar as operações de configuração. A implementação desta invenção requer uma base de informações relevante para armazenar um conjunto de dados usados para lidar com tarefas diferentes.

[014] O método de alteração de interface automatizada permite a configuração rápida da interface do usuário durante as operações de implementação ou execução do sistema, proporcionando capacidades flexíveis e amigáveis de configuração da interface pelo próprio usuário/usuária e possui capacidades de configuração de interface com base em parâmetros de elementos separados criados no sistema.

[015] O método apresentado nesta invenção é destinado a ativar / desativar quando da implementação dos blocos funcionais inteiros sem alterar o aplicativo em si, ou seja, sem excluir ou modificar quaisquer partes do código do software do sistema.

[016] Por exemplo, o desenvolvedor pode destacar a capacidade de trabalhar com propriedades adicionais do objeto, por exemplo, as propriedades "Mercadorias", como um elemento de sistema separado. Neste caso, se o desenvolvedor desativa esse recurso, todas as capacidades relacionadas com propriedades "Mercadorias" adicionais serão escondidos na interface do sistema. Com as configurações especificadas, o sistema automaticamente define-se em um novo estado e faz a interface mais fácil de se compreender, livre de elementos desnecessários e conveniente para trabalhar.

[017] O método apresentado na primeira versão compreende a criação de um elemento de sistema separado preliminarmente em fase de desenvolvimento (Fig. 1-120), especificando seus elementos dependentes que o compõem (1-1 Fig. 10), e salvar dito elemento separado no dispositivo de armazenamento. Também é necessário especificar neste elemento quais outros elementos do sistema este elemento influencia. Diferentes Elementos do sistema, como atributos, os comandos podem ser parte de elementos do sistema separados. Os atributos são catálogos campos de documentos que são exibidos no formulário para o usuário ler ou completar. O comando é um objeto de configuração usada pelo desenvolvedor para descrever ações a serem executadas pelo usuário.

[018] Em seguida, um elemento especial é formado (Fig. 1-130) para armazenar valores do elemento de sistema separado (Fig. 1-120). Este elemento especial (Fig. 1-130) tem uma propriedade necessária para ser especificado - a propriedade de armazenamento, para o qual você pode selecionar um objeto para servir como fonte de valores de elementos. Você pode selecionar, por exemplo, objetos tais como constantes, atributos de catálogo, e recursos de registrar de informações. Valores de armazenagem constantes que mudam raramente, por exemplo, nome da organização, cadastro de contribuinte, nome do diretor etc. Catálogos são ferramentas utilizadas para se trabalhar com uma lista de elementos homogêneos de dados. Qualquer catálogo tem dois atributos necessários - "Código" e "Nome" - que são criados automaticamente. Outros atributos do catálogo podem conter outras informações além do nome e deve ser determinado

para cada catálogo específico que está sendo criado. Registos de informação destinam-se a armazenar informações implementadas por uma combinação de dimensões. Recursos de registro são dados quantitativos ou integrais, armazenados no registro. Para gerir a interface, apenas os elementos separados, cujos valores são armazenados nos atributos do tipo booleano podem ser utilizados.

[019] Na fase de implementação ou execução (Fig. 1-200), o aplicativo cria um valor especial do referido elemento de sistema (Fig. 1-210). Na fase de execução do aplicativo (Fig. 1-300), o elemento separado criado no sistema é baixado (fig. 1-120), o valor do elemento separado é obtido a partir do elemento de valor especial de armazenamento (Fig. 1-130), e elementos de interface do usuário (Fig. 1-310) correspondentes aos elementos de composição dependentes (Fig. 1-110) são reconfigurados com base no valor obtido do elemento especial (Fig. 1-210). Em um caso particular, o elemento de interface do usuário (Fig. 1-310) que depende de vários elementos separados ou é parte de vários elementos distintos (Fig.1-120) será exibido se assim for permitido por, pelo menos, um daqueles elementos separados (Fig.1-120).

[020] A primeira versão da presente invenção pode ser exemplificada por uma tarefa, quando for necessário, dependendo do ambiente de uma implementação em particular, para proporcionar a desativação da capacidade de contabilidade do cliente por divisões que servem os referidos clientes, de modo que a “divisão” de campo não seja apresentada sob a forma de documento, quando o documento é manuseado. Para este fim, um elemento separado (Fig. 1-120.) – o elemento “Contabilidade da Divisão” - deve ser criado. Um elemento de sistema dependente (Fig. 1-110) (por exemplo, o "Recibo do Cliente" atributo documento) deve ser especificado no elemento separado componente. Em seguida, um elemento especial (Fig. 1-130), constante “Contabilidade da Divisão”, deve ser formada para armazenar fontes do tipo de valores de elementos do sistema do tipo booleana. O valor de elemento especial (1 Fig. -210) Deve ser definido como “Falsa” na fase de implementação ou execução (Fig. 1-200 e Fig.1-300), isso desativa o elemento separado “Contabilidade da Divisão” (Fig. 1-120) e o sistema reconfigura automaticamente a representação de todos os elementos de interface (fig. 1-310) compondo o elemento separado (Fig. 1-120), tal como, por exemplo, campos, comandos,

elementos de relatório, e neste caso em particular, a interface do usuário oculta o campo “da Divisão” campo.

[021] No que diz respeito à segunda versão da presente invenção, o valor do elemento separado (Fig. 2-120) depende de alguns parâmetros. Para implementar essa dependência, elementos especiais, parametrizando valores de elementos do sistema separados, são utilizados.

[022] Este método compreende a criação de um elemento de sistema separado (Fig. 2-120) preliminarmente na fase de desenvolvimento (Fig. 2-100), especificando seus elementos dependentes que o compõem (Fig. 2-110), e salvar tais elementos separados no dispositivo de armazenamento. Em seguida, o elemento de valores de parametrização (Fig. 2-140) do elemento de sistema separado (Fig. 2-120) é criado.

[023] Esse elemento deve especificar um conjunto de objetos cujos valores vão determinar a seleção de valores de elementos separados. A lista de objetos disponíveis inclui os catálogos e as dimensões do registo de informações. Neste caso, cadastram-se dimensões que mostram quais pontos de vista devem ser usados para armazenar as informações. Um catálogo da lista total de catálogos e uma dimensão de cada registo informações podem ser selecionados por cada valor de parâmetro na lista. Por exemplo, se um elemento separado (Fig. 2-120) é armazenado no atributo catálogo, em seguida, o parâmetro deverá incluir uma conexão a um elemento de catálogo particular. Se o elemento separado (Fig. 2-120) é armazenado no recurso de registo de informações, devem ser especificados os valores de todas as dimensões de registo. Neste caso, todas as dimensões devem ser caracterizadas pelo seu parâmetro. Parâmetros tornam possível a criação de elementos de rede separados, com um resultado que não tenha sido definido apenas uma vez na fase de aplicação, mas que pode variar de acordo com os dados armazenados nas bases de dados, conforme o parâmetro do elemento de sistema separado (Fig. 2-120).

[024] As próximas etapas incluem a formação de um elemento destinado a armazenar combinações de valores de parâmetros, com valores do elemento de sistema separado (Fig. 2-130) e entrando combinações de valores de parâmetros com os valores dos

elementos separados do sistema (Fig. 2-210) na fase de implementação do aplicativo ou sua execução (Fig. 2-200).

[025] Em seguida, o valor do parâmetro (Fig. 2-310) na fase de execução do aplicativo (Fig. 2-300) deve ser definido. O valor do elemento separado (Fig. 2-320) que corresponde ao valor do parâmetro específico (Fig. 2-310) é selecionado a partir do elemento de armazenamento de combinação de valores de parâmetros com os valores do elemento de sistema separado (Fig. 2-130). Em seguida, os elementos da interface do usuário (Fig. 2-330) conectados com elementos de composição dependentes (Fig. 2-110) são reconfigurados com base no valor elemento sistema separado (Fig. 2-120). Se nem todos os parâmetros são especificados para um elemento separado (Fig. 2-120) do tipo booleano, em seguida, a operação “Ou” para todos os valores com parâmetros não especificados é realizada. Por exemplo, se um elemento separado (Fig. 2-120) é armazenado nas informações de registro com a dimensão “Organização” e dimensão “Armazém”, e somente a dimensão “Organização” é especificada, então o valor do elemento separado (Fig. 2-120) será “Verdadeiro”, desde que o valor do elemento separado (Fig. 2-120) durante pelo menos um dos armazéns listados na dimensão “O Armazém” seja “Verdadeiro”.

[026] A próxima versão da presente invenção pode ser exemplificada por uma tarefa quando é necessário exibir ou ocultar o campo “Taxa IVA” (Imposto de Valor Agregado) em documentos, dependendo da organização para a qual a entrada de mercadorias é registrada. Se a organização mantém contabilidade fiscal em separado o campo “Taxa IVA” deve ser exibido para permitir que o usuário escolha um comum (padrão) ou qualquer outro (por exemplo, redução) da taxa. Se a organização não mantiver contabilidade separada, este campo deve ser escondido com um padrão básico “Taxa IVA” a ser usado. Para atingir este objetivo, o método exige a criação de um elemento separado - ou seja, o elemento “Contabilidade independente IVA” - (Fig. 2-120) no sistema, especificando seus elementos que compõem dependentes, e salvando o (Fig. 2-110) elemento criado no dispositivo de armazenamento. O próximo passo é criar o elemento “Organização” (Fig. 2-140) de parametrizando valores do elemento separado (Fig. 2-120), e formando o elemento para armazenar combinações de valores de

parâmetros com os valores do elemento separado (Fig. 2-130). Assim, o valor do elemento independente (Fig. 2-120) será armazenado no atributo catálogo “O Catálogo.Organizações.Atributo. Contabilidade independente de” do tipo booleano. Na fase da implementação do aplicativo ou execução (Fig. 2-200 e Fig. 2-300), o valor “Organização” do parâmetro está configurado (Fig. 2-310), e os valores correspondentes “Falso” ou “Verdadeiro” são selecionados. Então, se a organização não mantém uma contabilidade fiscal em separado o campo “Taxa IVA”, será escondido, mas se sim na outra via, este campo será exibido. Semelhante ao primeiro exemplo de realização, em um caso particular, o elemento de interface do usuário que depende de vários elementos separados ou é parte de vários elementos distintos será exibido se assim for permitido por, pelo menos, um desses elementos separados (Fig. 2-120).

[027] Assim, os métodos de alteração de interface automatizada oferecida pelas formas de realização desta invenção melhoram o conjunto de ferramentas tecnológicas e permite configurar a interface de usuário de forma flexível na sua fase de implementação e execução.

REIVINDICAÇÕES

1. Método de alteração de interface de usuário automatizada nas etapas de desenvolvimento, implementação e execução de aplicativos de negócios, **caracterizado por** compreender as seguintes etapas:

- criar pelo menos um elemento de sistema separado no estágio de desenvolvimento (Fig. 1-120) do referido aplicativo;

- especificar elementos dependentes de composição correspondentes a pelo menos um elemento de sistema separado, os elementos dependentes são selecionados pelo desenvolvedor a partir de uma lista então atual de elementos disponíveis dentro do aplicativo;

- criar um elemento de armazenamento (Fig. 1 - 130) para armazenar correspondências de valores de parâmetros e valores do elemento de sistema separado no estágio de desenvolvimento do aplicativo, em que os valores de cada um dos parâmetros e do valor de elemento de sistema separado são armazenados em campos no banco de dados;

- criar pelo menos um elemento de parametrização (Fig. 2 - 140) correspondente a um parâmetro no estágio de desenvolvimento de aplicativo;

- inserir correspondências dos valores de parâmetro (Fig. 2 - 200) com valores do elemento de sistema separado na fase de implementação do aplicativo;

- em resposta a uma mudança na configuração do usuário final, baixar um elemento de parametrização e definir seu valor de parâmetro no estágio de execução do aplicativo (Fig. 2 - 300);

- baixar o elemento de sistema separado (fig. 1-120) no estágio de execução do aplicativo e obter o valor do elemento de sistema separado a partir da correspondência armazenada do valor de parâmetro com o valor do elemento de sistema separado; e

- reconfigurar elementos de interface do aplicativo correspondentes aos elementos dependentes (Fig. 2 - 310), que são mostrados ao usuário final, com base nos valores obtidos do elemento de sistema separado no estágio de execução do aplicativo, em que os elementos de interface de aplicativo incluem campos, comandos e elementos de relatório.

2. Método, de acordo com a reivindicação 1, **caracterizado** pelo fato de que o elemento de interface de aplicativo, que depende de vários elementos de sistema separados, é exibido, se for permitido por pelo menos um desses elementos de sistema separados (Fig. 2 – 120).

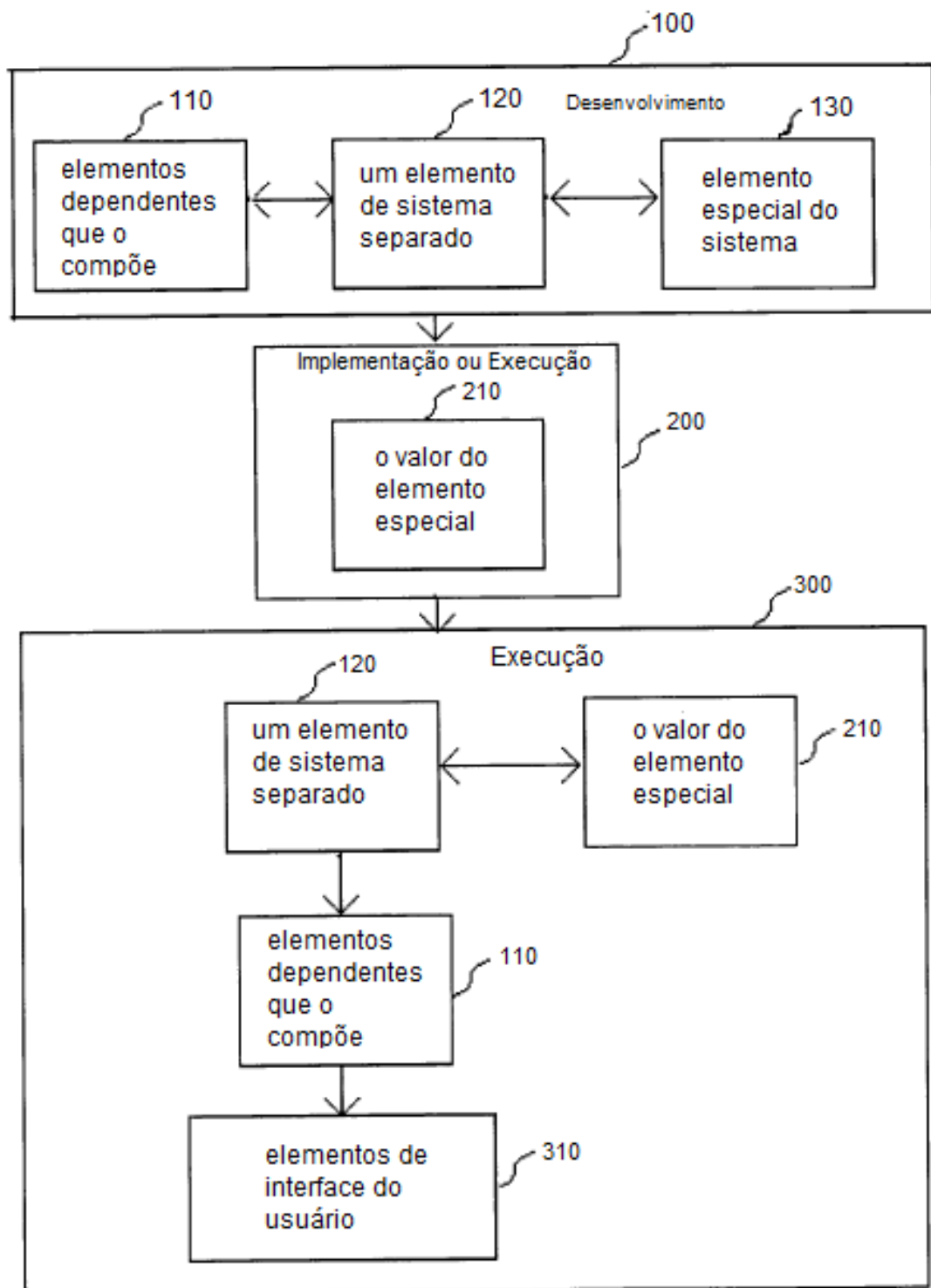


FIGURA 1

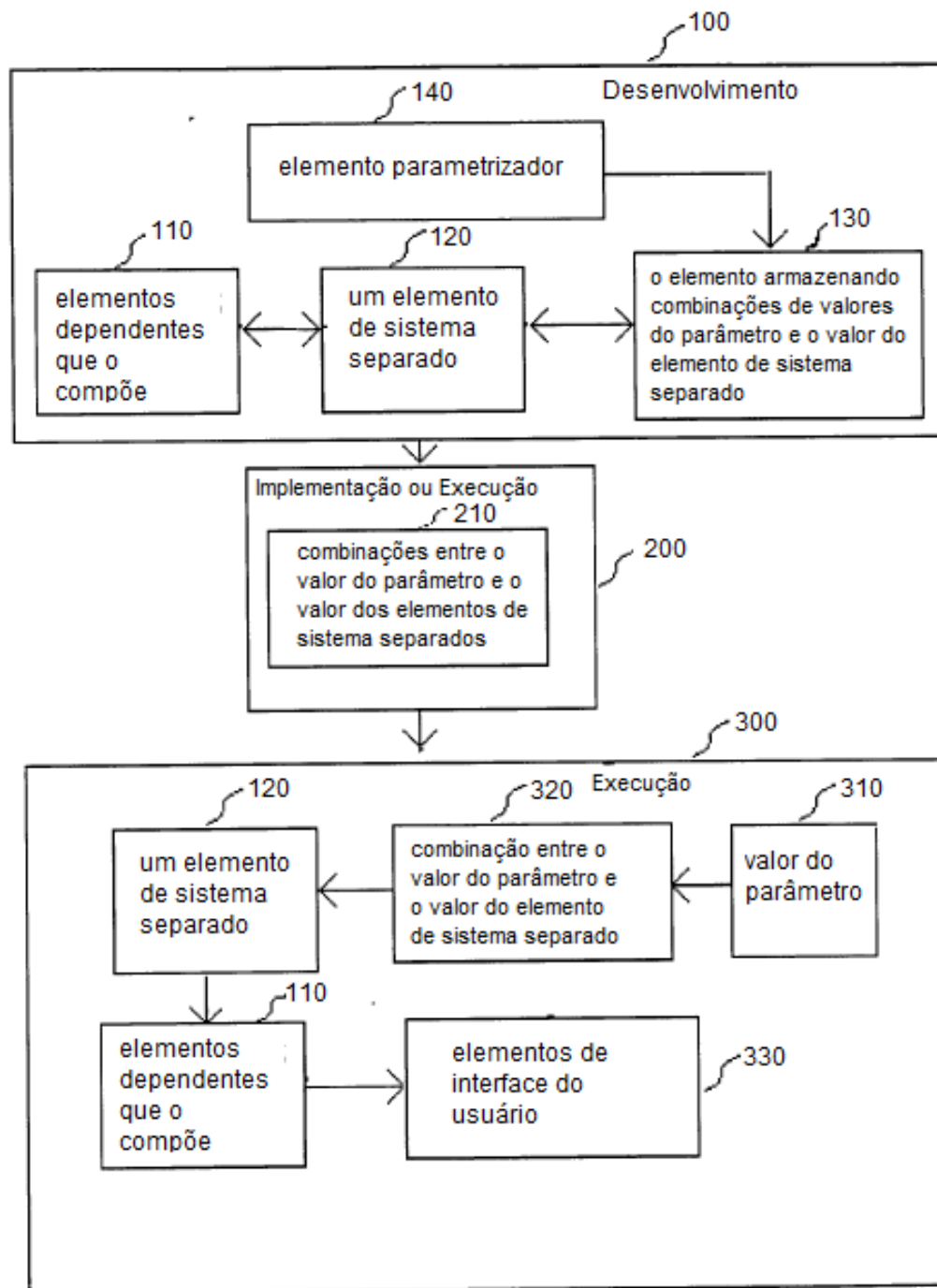


FIGURA 2