

A SYSTEM, METHOD AND COMPUTER READABLE MEDIUM FOR CONFIGURING AND RENDERING DYNAMIC FIELDS AND DYNAMIC OPTIONS IN A DYNAMIC FORM

TECHNICAL FIELD

[0001] The present disclosure relates generally to a system, method and computer readable medium for configuring and rendering dynamic fields and dynamic options in a dynamic form. Embodiments of the disclosure are related to a system, method and computer readable medium for configuring dynamic fields and options in the dynamic form and rendering dynamic fields and options based on rendering conditions.

BACKGROUND

[0002] Web-based forms serve an integral role in organizing information flow for today's business applications and offer significant productivity and efficiency gains as a means for data collection. Static forms are a type of web-based forms that are rendered once and displayed on the client or applicant and cannot be re-rendered in response to the interaction of the client or applicant. Dynamic Forms, on the other hand, are a type of web-based forms that generate relevant dynamic fields and options in real time, in response to the request, input or interaction of the client or applicant. Dynamic forms are rendered on the applicant or client dynamically and react on client events and interactions thereby capable of re-rendering on the client multiple times. Dynamic forms are also capable of saving the client's applicant's progress and pre-populating data to reduce errors.

[0003] The growing importance of dynamic forms has been acknowledged in all data-oriented fields and industries, to enable effective compilation and analysis of data, for example, banking and financial services, education sector, automobile, manufacturing industry, oil and gas, real estate, logistics, IT, healthcare, government and non-profit organizations, insurance and so on.

[0004] In the existing solutions, most of the dynamic form fields and options are configured manually through front-end technologies, by setting the conditions for

different scenarios to determine which fields and options are to be rendered. Therefore, in the existing solutions for configuring dynamic fields and options, the complete information for all the fields and the options to be rendered along with the conditions to render a specific dynamic field or option must be known before the final code implementation. Such methods are complicated and time consuming especially when the conditions are complex and there are a large number of dynamic fields and options. Accordingly, if any addition or alteration is to be made to the conditions or the fields or options to be rendered, the codes will need to be changed accordingly, thereby making the existing solutions inevitably highly time-consuming and requiring repetitive development and testing rendering relations.

[0005] In existing solutions, complex conditions are programmed manually with lots of time and effort to render a dynamic field or option. Any change in the business logic results in a requirement to change the coding used to render the dynamic field by the developer. The reason behind this lengthy configuration process is the tight coupling development mode where business logic is written as a part of the code. Such methods are not only complicated, time consuming, tedious and mundane, but also incur repetitive efforts in significantly amending the underlying codes especially when the conditions for the dynamic form fields or options are complex and there are a large number of dynamic fields and options.

SUMMARY

[0006] The following summary is provided to facilitate an understanding of some of the innovative features unique to the disclosed embodiment and is not intended to be a full description. A full appreciation of the various aspects of the embodiments disclosed herein can be gained by taking into consideration the entire specification, claims, drawings, and abstract as a whole.

[0007] The present disclosure relates generally to a system, method and computer readable medium for configuring and rendering dynamic fields and dynamic options in a dynamic form. Embodiments of the disclosure are related to a system, method and computer readable medium for configuring dynamic fields and options in the dynamic

form and rendering dynamic fields and options based on rendering conditions.

[0008] The system, method and computer readable media disclosed in the present application are advantageous in several ways including, but not limited to, providing a solution for configuring and rendering a dynamic form with dynamic fields and options within or across multiple tabs, simplified coding and easy and simple development of new dynamic fields and options without re-coding overall execution logic, reduced repetitive efforts for coding in case of addition or change in field or option conditions, reduced coding efforts on account of de-coupling of business logic at coding stage, circumventing hard-coding by providing configuration boards, dynamic visualization of the form thus making it convenient for business users to configure a desired dynamic form independently and quickly, enabling flexible configuration of dynamic form which allows a business user to choose the fields to be displayed in the dynamic form and also change the sequence of the fields and options by using drag and drop, enabling flexible configuration of each field and option which allows a business user to define the attributes such as field type, data type, regex and render when and so on, allowing business users (function or platform owners) to set different permission(s) and role(s) for different responsible personnel of the business user such as, action level permissions, such as permissions to view-only, create, edit or delete, enabling configuration of self-defined fields and options which can be used to auto render a dynamic field or option when a certain field's or option's value meets certain requirements, enabling configuration of dynamic fields and options without any limitation on number, rows or the like, enabling the configured dynamic form to be stored in formats that enable easy and hassle-free syncing and adaptation to a third-party system for rendering, resulting in reduced repetitive efforts and improved efficiency, resulting in improved flexibility and scalability by making dynamic fields configurable and minimizing errors/bugs by reducing coding changes and re-deployment.

[0009] It is therefore, one aim of the disclosed embodiments, to provide a system, method and computer readable medium for configuring and rendering dynamic web-based fields and options in a dynamic form that is easily configurable and adapts to a client's input to dynamically generate new information fields. Embodiments of the present invention provide a solution of programming a smart fillable online form with

configurable dynamic fields or options regardless of being within one tab or across multiple tabs, in programming formats, such as JSON format, which can also be used to send to any third-party system for dynamic form rendering.

[0010] It is therefore, one aim of the disclosed embodiments, to provide a system, method and computer readable medium for configuring and rendering dynamic fields and dynamic options in a dynamic form such that it allows a non-programmer user, such as a business user to customize the form and the rendering condition for each field or option dynamically and easily, without creating a requirement for change in the underlying codes. Embodiments of the present invention allow the business user to amend the dynamic fields and options, such that the changes made do not affect the main codes and coding, and thereby reducing repetitive work and improving the efficiency.

[0011] It is therefore, one aim of the disclosed embodiments, to provide a system, method and computer readable medium for configuring and rendering dynamic fields and dynamic options in a dynamic form such that the coding is decoupled from the business logic, thereby turning the code implementation of dynamic fields and options rendering to page configuration. This in turns makes the coding easier and can reduce the labour cost and time. Embodiments of the present invention allow business users (function or platform owners) to set different level of permissions and roles to different responsible personnel of the business user, for example, action level permission can be view-only, create, edit, delete permission. Also, by making the dynamic fields and options configurable, the present invention can make the system more flexible and improve the scalability of the system.

[0012] In a first aspect of the present disclosure, a system for configuring and rendering a dynamic field and a dynamic option in a dynamic form is disclosed. The system comprises, a visual representation means for visually displaying one or more of: a field, an option and a form, such that the visual representation means displays one or more of a fields configuration board or an options configuration board or a forms configuration board; and an execution unit comprising at least one configuration program and at least one storage unit, wherein the configuration program enables the

configuration of the fields on the fields configuration board and the configuration of the options on the options configuration board; wherein the configuration program enables the configuration of the dynamic field on the fields configuration board with at least one rendering condition and at least one logical operator; wherein the dynamic field is rendered when the rendering condition is satisfied; wherein the configuration program enables the configuration of the dynamic option on the options configuration board with at least one rendering condition and at least one logical operator; wherein the dynamic option is rendered when the rendering condition is satisfied; wherein the configuration program enables the configuration of the dynamic form on the forms configuration board with one or more of the dynamic field and the dynamic option; and wherein the storage unit stores for each field, a field code and a value corresponding to the field.

[0013] According to an embodiment in conjunction to the first aspect of the present disclosure, the rendering condition uses the field code and the value of the field.

[0014] According to an embodiment in conjunction to the first aspect of the present disclosure, the dynamic field is configured and rendered on one or more page tabs.

[0015] According to an embodiment in conjunction to the first aspect of the present disclosure, the field code and the value are stored in the storage unit when the field is configured.

[0016] According to an embodiment in conjunction to the first aspect of the present disclosure, the dynamic field is rendered based on the rendering condition and the logical operator based on an other field.

[0017] According to an embodiment in conjunction to the first aspect of the present disclosure, the other field is with the option.

[0018] According to an embodiment in conjunction to the first aspect of the present disclosure, the fields configuration board for each field, enables defining one or more attributes, such that the attributes include a field type, a data type, a regex, a field

code, a parent code, a maximum length or a description of the field.

[0019] According to an embodiment in conjunction to the first aspect of the present disclosure, the options configuration board for each option enables defining one or more attributes, such that the attributes include a code value, a sequence or an options group code.

[0020] According to an embodiment in conjunction to the first aspect of the present disclosure, the configuration program enables assigning the values to the attributes, defining the field code, setting the options and setting the rendering conditions and the logical operators to the options.

[0021] According to an embodiment in conjunction to the first aspect of the present disclosure, the storage unit stores the dynamic form with the fields and the options configured using the configuration program.

[0022] According to an embodiment in conjunction to the first aspect of the present disclosure, the dynamic form is synchronized and rendered to a third-party system.

[0023] According to an embodiment in conjunction to the first aspect of the present disclosure, the fields configuration board is a fields configuration page.

[0024] According to an embodiment in conjunction to the first aspect of the present disclosure, the options configuration board is an options configuration page.

[0025] In a second aspect of the present disclosure, a method for configuring and rendering a dynamic field and a dynamic option in a dynamic form is disclosed. The method comprises: configuring, by a fields configuration board in a visual representation means using a configuration program, a field in the dynamic form; configuring, by an options configuration board in the visual representation means using the configuration program, an option in the dynamic form; configuring, by the fields configuration board using the configuration program, the dynamic field in the dynamic form with at least one rendering condition and at least one logical operator;

and configuring, by the options configuration board using the configuration program, the dynamic option in the dynamic form with at least one rendering condition and at least one logical operator.

[0026] According to an embodiment in conjunction to the second aspect of the present disclosure, the method further comprises: initializing, by an execution unit, one or more fields, options and forms in an order of configuration; registering, by the execution unit, each field in a storage unit with a field code and a value corresponding to the field along with the rendering condition and the logical operator; checking, by the execution unit, the rendering condition and the logical operator of the dynamic field or the dynamic option to be rendered on other fields or options; and determining, by the execution unit, renderability of the dynamic field based on the satisfaction of the rendering condition and the logical operator.

[0027] According to an embodiment in conjunction to the second aspect of the present disclosure, the method further comprises: receiving, by the execution unit, the value of one or more of the fields or the options in the dynamic form; informing, by the execution unit, the value of the fields or the options to the storage unit; detecting, by the execution unit, an input or change in the value of the fields or the options; checking, by the execution unit, the rendering condition and the logical operator of the dynamic field or the dynamic option to be rendered on other fields or options; and determining, by the execution unit, renderability of the dynamic field based on the satisfaction of the rendering condition and the logical operator.

[0028] According to an embodiment in conjunction to the second aspect of the present disclosure, the method further comprises: sending, by the execution unit, an instruction to render the dynamic field or the dynamic option to the visual representation means; and rendering, by the visual representation means, the dynamic field or the dynamic option in the dynamic form.

[0029] According to an embodiment in conjunction to the second aspect of the present disclosure, the method further comprises: receiving, by a third-party system, the dynamic form with one or more of the dynamic field or the dynamic option;

decoding, by the third-party system, configuration of the dynamic form; obtaining, by the third-party system, the dynamic form with one or more of the dynamic field or the dynamic option with the rendering condition and the logical operator; and rendering, by the third-party system, the dynamic form.

[0030] In a third aspect of the present disclosure, a computer readable medium storing a program comprising instructions that, when executed by a system, cause the system to perform a method for configuring and rendering a dynamic field and a dynamic option in a dynamic form. A computing device comprising the computer readable medium can perform the method for configuring and rendering a dynamic field and a dynamic option in a dynamic form.

BRIEF DESCRIPTION OF THE DRAWINGS

[0031] FIG. 1 is a block diagram depicting an example system **100** for configuring a dynamic field and a dynamic option in a dynamic form, in accordance with the disclosed embodiments;

[0032] FIG. 2 is a non-limiting schematic diagram that illustrates an example system **200** for rendering a dynamic form with dynamic fields and options at a computing device, in accordance with the disclosed embodiments;

[0033] FIG. 3 is a schematic diagram that illustrates an example options configuration board **300** for configuring options for one or more fields of a dynamic form, in accordance with the disclosed embodiments;

[0034] FIG. 4 is a schematic diagram that illustrates an example field configuration **400** at a fields configuration board, in accordance with the disclosed embodiments;

[0035] FIG. 5 is a schematic diagram that illustrates another example of a field configuration at a fields configuration board **500** for configuring a dynamic field with rendering conditions and logical operator, in accordance with the disclosed embodiments;

[0036] **FIG. 6A** is a schematic diagram that illustrates rendering of an example dynamic form **610** for selecting an option from an option list at a computing device, in accordance with the disclosed embodiments;

[0037] **FIG. 6B** is a schematic diagram that illustrates rendering of an example dynamic form **620** showing an option selected from an option list, in accordance with the disclosed embodiments;

[0038] **FIG. 6C** is a schematic diagram that illustrates rendering of an example dynamic form **630** showing a dynamic field displayed upon selecting an option from an option list, in accordance with the disclosed embodiments;

[0039] **FIG. 7** is a flow diagram that illustrates an example configuration method **700** performed at visual representation means, in accordance with the disclosed embodiments;

[0040] **FIG. 8** is a flow diagram illustrating an example method **800** for rendering a dynamic form with dynamic fields and options, performed at the execution unit, in accordance with the disclosed embodiments;

[0041] **FIG. 9** is a flow diagram that illustrates an example method **900**, performed at an execution unit, for rendering a dynamic form, in accordance with the disclosed embodiments; and

[0042] **FIG. 10** is a flow diagram that illustrates an example method **920**, performed at a third-party system, for rendering a dynamic form, in accordance with the disclosed embodiments.

[0043] The summary above, as well as the following detailed description of illustrative embodiments, is better understood when read in conjunction with the appended drawings. For the purpose of illustrating the present disclosure, exemplary

constructions of the disclosure are shown in the drawings. However, the disclosure is not limited to specific methods and instrumentalities disclosed herein. Moreover, those in the art will understand that the drawings are not to scale. Wherever possible, like elements have been indicated by identical numbers.

DETAILED DESCRIPTION OF ILLUSTRATIVE EMBODIMENTS

[0044] The particular configurations discussed in the following description are non-limiting examples that can be varied and are cited merely to illustrate at least one embodiment and are not intended to limit the scope thereof.

[0045] The present disclosure allows a business user to customize a dynamic form and rendering conditions for each field or option, based on its business requirements, without having to amend the underlying codes. Further, the present invention also allows the business user to amend the fields and options, such that the changes made do not affect the underlying codes and programming, and thus reduce repetitive work and improves efficiency.

[0046] The present disclosure provides flexible configuration of dynamic form by allowing a business user to choose the field to be displayed in the dynamic form. The business user is further allowed to change the sequence of the fields in the dynamic form by using drag and drop. The present disclosure further enables flexible configuration of each field and option by allowing a business user to define its attributes, such as field type, data type, regex, “render when” and so on. One of a significant advantage provided by the present disclosure is allowing a business user to self-define the rendering conditions for various dynamic fields and options.

[0047] Various non-limiting benefits of the present disclosure includes, providing solution for configuring and rendering dynamic fields and options of dynamic form regardless of within one tab or across multiple tabs, enabling each field or option to be a dynamic one depending on a render condition without any restrictions, providing a dynamic form that are easily configurable and adaptable to a client’s input to dynamically generate new information fields, enabling a non-programmer user such

as a business user to customize its form, the fields and associated options, and the rendering conditions and logical operator of each field and option dynamically without having to change the underlying codes.

[0048] The present disclosure also, decouples the business logic from coding by converting code implementation of dynamic fields and options rendering to page configuration. The system, method and computer readable medium provided in the present disclosure allows business users (function or platform owners) to set different permissions for different business users such as, action level permissions which can be view-only, create, edit or delete. Implementations of the system, method and computer readable medium provided in the present disclosure results in reduced repetitive efforts and improved efficiency, improved flexibility and scalability by making the dynamic fields configurable, minimized errors/bugs by reducing the requirements of system changes and re-deployments and simplified coding by providing management boards for dynamic form and field and option configuration and therefore, eliminating the need of hard-coding.

[0049] A dynamic form can have different tabs for different classifications of information to be collected. For example, in banking or financial services, a dynamic form can have separate tabs such as 'application info', 'personal information', 'business info', 'business address' and so on. Further, a dynamic form can have different types of fields, including without limitation, textboxes, dropdowns, checkboxes, radio buttons and so on. A field relates to any type of user interface control via which data input can be carried out. A field in a dynamic form can be a condition field or a dynamic field. A condition field requires input data by way of input text, for example, name, age, gender and so on. A dynamic field is rendered in a dynamic form based on an input received in a previous condition field of the dynamic form. Thus, rendering of a dynamic field in a dynamic form is contingent on the rendering conditions of the dynamic field being satisfied. For example, in an academic application form, a dynamic field for "thesis details" may only be rendered if the applicant responds with a "yes" option to the field "undertaken a thesis project". In the example, if the applicant would have responded with a "no" option to the field "undertaken a thesis project", the dynamic field "thesis details" would not be rendered

to the applicant. In another example, a dynamic field requiring an entry for 'spouse name' may be configured such that it will be rendered only if option 'married' is selected in a previous condition field 'marital status'. Certain fields can have a list of option values, out of which one can be selected. Like dynamic fields, a list of dynamic options can also be rendered based on a selection of an option or value input in a field from another list of options provided in the dynamic form, for instance, a conditional field "Country" may be configured in a dynamic form, with dropdown options of "USA", "Singapore", "China", and so on, and a dynamic field "Province" may be configured with dropdown options of "Beijing", "Shanghai" and "Guangdong", and so on. Further, a dynamic field "City" may be configured with dropdown options of "Guangzhou" and "Shenzhen", and so on. The dynamic options "Guangzhou" and "Shenzhen" may be configured to be rendered only when option "China" is selected from the options for field "Country" and option "Guangdong" is selected from the options for field "Province".

[0050] The present disclosure provides a solution which helps a business user, in programming a smart fillable online form with configurable dynamic fields and options, regardless of within one tab or across multiple tabs, in a format which can also be used to send to any third-party system for dynamic form rendering, for example, JSON format. Such dynamic forms as provided by the present disclosure, for example, may be used for collecting data from customers, to aid in the assessment of their suitability for certain financial transactions, such as loan applications.

[0051] The system, method and computer readable medium of the present disclosure enables configuring and rendering of dynamic fields and dynamic options in a dynamic form; and allows a user to configure one or more dynamic fields and options, with attributes such as field type, data type, regex, "render when" and so on, using one or more configuration boards. A rendering condition to render a dynamic field or a dynamic option may be set in such a manner that when a value in another field or option of the dynamic form changes, the dynamic field or dynamic option is rendered accordingly. Therefore, a business user can define rendering conditions using one or more field code and value pairs with a logical operator. Some implementations of the present invention enable a business user to comfortably customize an online business form and set rendering conditions relevant to the

requirements of its business and render certain fields dynamically when the set rules are satisfied, without having to amend the underlying codes.

[0052] FIG. 1 is a block diagram depicting an example system **100** for configuring a dynamic field and a dynamic option in a dynamic form, in accordance with the disclosed embodiments. The system **100**, for example, comprises an execution unit **106** and at least one Graphic user interface (GUI) or visual representation means **102** of a computing device. The execution unit **106** comprises a configuration program **104** and a storage unit **108**. The configuration program **104** generates one or more of a fields configuration board **110**, an options configuration board **112** and a forms configuration board **114**, collectively referred to as the management boards. The fields configuration board **110** is the board used to manage and configure dynamic and conditional fields in a dynamic form. The fields configuration board **110** can comprise, for example, various attributes such as parent, code, field type, data type, option group, unit group, maximum length, regular expression, description, logical operator and render conditions.

[0053] The options configuration board **112** is the board used to manage and configure dynamic and conditional options in a dynamic form. The options configuration board **112** can comprise, for example, various attributes such as code, sequence, rendering conditions and operation. The forms configuration board **114** is the board used to manage and configure the dynamic form.

[0054] A dynamic field and option can be configured using the configuration program **104**, on the fields configuration board **110** or the options configuration board **112**, respectively, with one or more rendering conditions and one or more logical operators (for example OR, AND, ANY).

[0055] According to an embodiment of the present disclosure, the fields configuration board is a fields configuration page and the options configuration board is an options configuration page.

[0056] Rendering conditions refer to the conditions on the basis of which a dynamic

field or option is rendered. Logic operator refers to a symbol or word used to connect two or more expressions such that the value of the compound expression produced depends only on that of the original expressions and on the meaning of the operator. Common logical operators include AND, OR, and NOT.

[0057] The rendering conditions are configured on the fields configuration board **110** using field value and value pairs. The logical operator can be used for one or more such fields and the value pair. The rendering condition and business requirement can also be configured for a field with options, depending on the business requirement.

[0058] The storage unit **108** in the execution unit **106** stores, including but not limited to, for each field, a field code and a value corresponding to the field; for each option, option corresponding to a field and its option list; various attributes of fields and options; rendering conditions and logical operators for each field and option, dynamic form with various tabs and pages and so on.

[0059] The GUI or visual representation means **102** visually represents or displays the dynamic fields and options configured at the execution unit **106** by way of the management boards **110**, **112** and **114**. The GUI **102** allows the business user to perform actions such as configuring a desired dynamic form independently and quickly, choosing the fields to be displayed in the dynamic form and also changing the sequence of the fields and options by using drag and drop, defining the attributes such as field type, data type, regular expressions (regex), render when and so on, setting different permission(s) and role(s) for different responsible personnel of the business user such as, action level permissions, such as permissions to view-only, create, edit or delete.

[0060] After configuration of dynamic forms with dynamic fields and options, the dynamic forms can be used by the business user, for example, in the banking and financial services, for collecting data from customers to aid in the assessment of their suitability for certain financial transactions, for example loan applications.

[0061] **FIG. 2** is a non-limiting schematic diagram that illustrates an example system **200** for rendering a dynamic form with dynamic fields and options at a computing

device, in accordance with the disclosed embodiments. As disclosed in **FIG. 2**, the system **200**, for example, comprises a visual representation means, for example, a Graphical User Interface (GUI) **202** at a computing device and an execution unit **220**.

[0062] During form rendering, all fields and options based on a configured form are initialized at the execution unit **220**. During initialization, each field (including field code along with rendering conditions, key value pairs for each field and logical operator), and each option (including code along with rendering conditions, value pairs for each field or option and logical operator) is registered to a storage unit **214** of the execution unit **220**, in a format that can be used conveniently for rendering at a third-party system, for example, JSON format. In some implementations, the storage unit **214** may be a repository.

[0063] The GUI **202** renders or displays to a client, the fields one by one, for example, Field A **204** and Field B **206**. If a field or option is dynamic and depends on other fields or options, the execution unit **220** will check based on the storage unit **214** and determine whether a certain field should be displayed or not. A rendered field, such as Field A **204** is also registered as Field A **210** to a listener **216** in the execution unit **220**. The listener **216** or event listener listens **222** at the storage unit **214**. An event listener or listener or event handler is a concept or function that waits for an event to occur. Listener concepts are programmed to react to an input or signal by calling the event's handler. In **FIG. 2**, for example, listener **216** listens **222** at the storage unit **214** and awaits a detected change **226** at the storage unit **214** regarding change of input in value of a field or option registered by the storage unit **214**. There are many event listener concepts in the industry which can be used to achieve the functionality of the listener **216**.

[0064] Once there is an input or change in the value of a field or option, the GUI **202** sends a signal to the execution unit **220**. For example, in **FIG. 2**, when the value of Field A **204** is input or changed, the GUI **202** sends a signal **208** to the execution unit **220**. The corresponding registered field in the execution unit **220**, for example, Field A **210**, informs **224** the storage unit **214** of the change or input in value at the respective field or option at the GUI **202**, for example, Field A **204**.

[0065] The listener **216** is listening **222** at the storage unit **214** and on detecting the change **226** at the storage unit **214**, regarding the change in value of a field, for example Field A **204**, the listener **216** will check and determine whether to render a respective dynamic field, for example Field B **212**, based on the change in the value of the field, for example Field A **210**. In case the dynamic field or option is to be rendered based on the rendering conditions and logical operator, the listener **216** will send a rendering signal **230** to the GUI **202**. Accordingly, the rendering signal **230** is sent to the GUI **202** and will be rendered on the dynamic form and visually represented by rendering of Field B **206**.

[0066] In an example, a condition field “Marital status” is configured as Field A **204** with dropdown options of “married”, “single”, “separated” and so on and a dynamic field “Spouse’s last name” is configured as Field B **206**. The rendering conditions and logic operator are such that the field “Spouse’s last name” will be rendered only if the value of the field “Marital Status” is either “married” or “separated”. In any of the other options, the field “Spouse’s last name” will not be rendered. Thus, according to **FIG. 2**, before form rendering, all the above-mentioned example fields and options are initialized at the execution unit **220**. During initialization, the above-mentioned fields are registered to a storage unit **214** in a programming format, for example, Field A **204** “Marital status” may be registered as “personMaritalStatusD” and Field B **206** “Spouse’s last name”, may be registered as “personSpouseLastName”, and the rendering conditions for dynamic field “Spouse’s last name” may be “personMaritalStatusD.married, personMaritalStatusD.separated|any”. In the above example, the dynamic form is rendered at the GUI **202** and each dynamic field is rendered one by one. When a value is input as “married” in Field A **204**, the GUI **202** informs the execution unit **220** at **208** and the corresponding registered field in the execution unit **220** Field A **210**, informs the storage unit **214** of the change in value at the respective field at the GUI **202**. The listener **216** is listening at the storage unit **214** and detects the change **226** and accordingly checks if Field B **212** is to be rendered or not. Since the rendering conditions and logical operator registered with Field B **212** in the storage unit are such that if the field value for Field A **210** is married, Field B **212** is rendered, the listener **216** will send a rendering signal **230** to the GUI **202**. Accordingly, field “Spouse’s last name” will be rendered on the dynamic form and visually represented by rendering of Field B **206** at GUI **202**. If the value at Field A **204**

is changed to “single”, the listener **216** will check and determine that field “Spouse’s last name” is not to be rendered at the GUI **202**.

[0067] FIG. 3 is a schematic diagram that illustrates an example options configuration board **300** for configuring options for one or more fields of a dynamic form, in accordance with the disclosed embodiments. As disclosed in **FIG. 3**, the options configuration board **300** is used to configure options in an options group **302** bearing a group code. The configured options in an options group **302** are listed in a code value column **304**. The code value column **304** in **FIG. 3** shows a list **303** of options, for example, “Single”, “Married”, “Widowed”, “Separated”, “Others”. The business user can set a desired viewing sequence of the options in a sequence row **306** by providing a sequence number to each option in the list **303**. For example, as disclosed in **FIG. 3**, in an example options configuration board **300**, the sequence number for the option list **303**, that is “Single”, “Married”, “Widowed”, “Separated”, “Others” is set as **1, 2, 3, 4** and **5** respectively. The business user can also set rendering conditions and logical operation details in “Render When” column **308** and operation details **310** for each option, if required. The option details such as, the option code, the field associated to the option and its attribute are stored in the storage unit at execution unit. The storage unit can be a repository unit as depicted in the **FIG.2**.

[0068] FIG. 4 is a schematic diagram that illustrates an example field configuration **400** at a fields configuration board, in accordance with the disclosed embodiments. As disclosed in **FIG. 4**, a business user can configure and manage each field at the fields configuration board by defining different attributes such as field type, data type, maximum length and so on. As disclosed in **FIG. 4**, the business user can set an options group **404** bearing a group code to a field bearing a field code **402** by selecting field type **406** as “Dropdown”. For example, as shown in the **FIG. 4**, the options group **404** having a group code “civilStatus_option” is set to the field having field code “personMaritalStatusD” **402**. The business user can also set values to various attributes of the field. The attributions are not limited to field type, field code, data type of the field, maximum length of the field, regex for the field and the description of the field. The field details such as, the field code, field value, the option associated to the field and its attribute are stored in the storage unit at execution unit. The storage unit

can be a repository unit depicted in the **FIG. 2**. Using the present invention, the condition fields and dynamic fields or options can be configured either within the same tab or across different tabs, with the sequence of condition fields first followed by dynamic fields/options.

[0069] **FIG. 5** is a schematic diagram that illustrates another example of a field configuration at a fields configuration board **500** at a fields configuration board for configuring a dynamic field with rendering conditions and logical operator, in accordance with the disclosed embodiments. As disclosed in **FIG. 5**, rendering conditions **510**, **512** and logical operator **506**, **508** can be configured for a dynamic field. Rendering conditions refer to the conditions on the basis of which a dynamic field or option is rendered. Logic operator refers to a symbol or word used to connect two or more expressions such that the value of the compound expression produced depends only on that of the original expressions and on the meaning of the operator. As disclosed in **FIG. 5**, logical operators for example, “ANY” **506** and “ALL” **508** can be set in the rendering condition. Common logical operators include AND, OR, and NOT. In the field configuration **500**, the dynamic field with field code **502** “personSpouseLastName” is configured with rendering conditions as “personMaritalStatusD.married” **510** and “personMaritalStatusD.seperated” **512** and the logical operator used is “ANY” **506**. The dynamic field is the field **502** with the field code “personSpouseLastName” and field type as “Input field”. The dynamic field **502** is displayed only when anyone of the options, for example “married” or “separated” is selected in the field “Marital status” with field code “personMaritalStatusD” and field type “dropdown”. The rendering condition details, such as, the option and its list, the logical operator used in the rendering condition, the dynamic field that to be rendered based on the rendering condition, are stored in the storage unit at execution unit. The storage unit can be a repository unit depicted in the **FIG. 2**.

[0070] In an example, **FIGS. 3, 4 and 5** can be used for a condition field “Marital status” whose code is “personMaritalStatusD” with dropdown options of “married”, “single”, “separated” and so on, and another field “Spouse’s last name” with code “personSpouseLastName” is a dynamic field. If the rendering configuration and logical operations applied to render the field “personSpouseLastName” is

“personMaritalStatusD.married, personMaritalStatusD.separated|any”. The dynamic field, “personSpouseLastName” is rendered only when the value input to the field “Marital status” with code “personMaritalStatusD” is either “married” or “separated”.

[0071] In another example, a conditional field “Country” may be configured in a dynamic form, with dropdown options of “USA”, “Singapore”, “China”, and so on, and a dynamic field “Province” may be configured with dropdown options of “Beijing”, “Shanghai” and “Guangdong”, and so on. Further, a dynamic field “City” may be configured with dropdown options of “Guangzhou” and “Shenzhen”, and so on.

The rendering configuration “Country.China,Province.Guangdong|all” may be configured and applied to options “Guangzhou” and “Shenzhen” for rendering the respective. Accordingly, the dynamic options of “Guangzhou” and “Shenzhen” are shown only when option “China” is selected from the options for field “Country” and option “Guangdong” is selected from the options for field “Province”.

[0072] **FIG. 6A** is a schematic diagram that illustrates rendering of an example dynamic form **610** for selecting an option from an option list **602** at a computing device, in accordance with the disclosed embodiments. **FIG. 6B** is a schematic diagram that illustrates rendering of an example dynamic form **620** showing an option “Single” **604** selected from the option list **602** at the GUI of a computing device, in accordance with the disclosed embodiments. **FIG. 6C** is a schematic diagram that illustrates rendering of an example dynamic form **630** showing an option “Married” **608** selected from the option list **602** at the GUI of a computing device, in accordance with the disclosed embodiments. In the example dynamic form disclosed in **FIGS. 6A, 6B and 6C**, a condition field “Marital status” is configured with dropdown options **602** of “married” **608**, “single” **604**, “separated” and so on. The rendering conditions and logic operator are such that a field “Spouse’s last name” will be rendered only if the value of the field “Marital Status” is either “married” or “separated”. In any of the other options, the field “Spouse’s last name” will not be rendered. Thus, as disclosed in **FIG. 6B**, when the option “Single” **604** is selected, the field “Spouse’s last name” is not rendered. However, as disclosed in **FIG. 6C**, when the option “Married” **608** is selected from the list **602**, the dynamic field “Spouse’s last name” **610** is dynamically rendered to the

customer, client or applicant.

[0073] At the time when the fields, the options and the forms are initialized during dynamic form rendering, the attributes and its rendering condition are also stored in the storage unit of the execution unit, for example in the repository. The execution unit can have a listener or event listener to listen at the storage unit by sending a signal to the storage unit. When a value of field or option is input or changed in a field at the dynamic form by the client, customer or applicant, the value of field or option is detected at the storage unit by the listener, the listener will determine whether to render a dynamic field or not based on the change in the value of the field. In **FIG. 6C**, the dynamic field **610**, for example the field “personSpouseLastName” is rendered only when the marital status of the person is “Married” or “Separated”. In **FIG. 6B**, the dynamic field **610**, for example the field “personSpouseLastName” is not rendered as the marital status of the person is “single”. Change in the value of the option is intimated to the storage unit and the listener can listen and detect this change at the storage unit to check whether that change in the value should render the dynamic field or not.

[0074] The present invention advantageously enables a business operator to customize the online business form and the content of each fields dynamically and easily, without having to change the underlying codes. In this regard, the present invention turns the code implementation of dynamic field or options rendering to page configuration, thus separating the business logic from coding stage. This in turns makes the programming easier and can reduce the labour costs, time and effort. Business users (function or platform owners) are able to set different permission(s) and role(s) for different responsible personnel of the business user such as, action level permissions, such as permissions to view-only, create, edit or delete. In addition, changes on the business logic no longer affect programming, thus reduce repetitive work and improving work efficiency. Also, by making the dynamic field configurable, the present invention can make the system more flexible and improve the scalability of the system. Further, configuration of self-defined fields and options is enabled by the present invention which can be used to auto render a dynamic field or option when a certain field’s or option’s value meets certain requirements. Additionally, since the

configured dynamic forms, fields and codes are stored in formats that enable easy and hassle-free syncing and adaptation to a third-party system for rendering, the same results in reduced repetitive efforts and improved efficiency, improved flexibility and scalability by making dynamic fields configurable and minimizing errors/bugs by reducing coding changes and re-deployment

[0075] FIG. 7 is a flow diagram that illustrates an example configuration method **700** performed at visual representation means, in accordance with the disclosed embodiments. Once, one or more of the fields configuration board, the options configuration board and the forms configuration board has been generated using the computer program, as at step **702**, the method **700** can be implemented, for example, by configuring the fields in a dynamic form using the fields configuration board on the visual representation means, for example, the GUI. The fields configuration board uses the configuration program at the execution unit that allows the business users to create fields, options, page tabs and forms. The configuration program at the execution unit also enables the business users at the visual representation means to define various attributes to each field and option with or without rendering conditions and logical operators. As at step **704**, the method **700** can be implemented, for example, by configuring the options in a dynamic form using the options configuration board on the visual representation means. Each group code, that is, the option name can have two or more options in a list. The business user can define the option code, option list, sequence of the option in the list and so on, using the options configuration board. Then, as at step **706**, the rendering conditions and the logical operators are configured for each dynamic field to enable field level dynamic rendering at the fields configuration board. Thereafter, at step **708**, the rendering conditions and the logical operators are configured for each dynamic option to enable option level dynamic rendering at the options configuration board. To configure fields, options, pages, tabs and forms, the invention can use drag and drop facility at GUI.

[0076] It should be noted that a dynamic field or a dynamic option is displayed to the client or applicant only when the rendering conditions for said dynamic field or dynamic option are satisfied.

[0077] **FIG. 8** is a flow diagram illustrating an example method **800** for rendering a dynamic form with dynamic fields and options, performed at the execution unit, in accordance with the disclosed embodiments. As disclosed in **FIG. 8**, the example method **800** can be implemented, for example, at step **802**, by initializing all the fields and options configured in a dynamic form. As soon as a new field or option is configured, it is initialized in the execution unit.

[0078] Then, at step **804**, each field (including field code along with rendering conditions, key value pairs for each field and logical operator), and each option (including code along with rendering conditions, value pairs for each field or option and logical operator) is registered to a storage unit of the execution unit, in a format that can be used conveniently for rendering at a third-party system, for example, JSON format. In some implementations, the storage unit may be a repository. Thereafter, at step **806**, checking by the execution unit whether a dynamic field or option depends on other fields or options, from the storage unit, and determining whether a certain field should be displayed or not at the visual representation means, as at step **808**. If, at step **810**, the rendering conditions are not satisfied, then the execution unit will continue to check for other fields and options at step **806**. However, if, at step **810**, a rendering condition is satisfied, a rendering instruction is sent to a visual representation means, for rendering the respective dynamic field or option as at step **812**. It should be noted that the example method **800** can be performed using, for example, a repository and a listener which can be used to listen to an input or change in the value of a field or option at the repository as received from the visual representation means. The listener can thereafter determine whether or not to render the dynamic field or option.

[0079] **FIG. 9** is a flow diagram that illustrates an example method **900**, performed at an execution unit, for rendering a dynamic form, in accordance with the disclosed embodiments. The method **900** can be achieved by using for example, a repository and a listener as described in the example system **200** in **FIG. 2**. The listener (event listener) listens at the storage unit of the execution unit and awaits a detected change at the storage unit regarding change of input in value of a field or option registered by the storage unit. An event listener or listener or event handler is a concept or function

that waits for an event to occur. Listener concepts are programmed to react to an input or signal by calling the event's handler. There are many event listener concepts in the industry which can be used to achieve the functionality of the listener for the embodiments of the present invention.

[0080] As disclosed in **FIG. 9**, the example method **900** can be implemented, for example, at step **902**, by receiving by the execution unit, an input or change in the value of the field or option at the visual representation means. Then, at step **904**, the execution unit informs the storage unit of the change or input in value at the respective field or option at the visual representation means. The listener is listening at the storage unit and on detecting the change at the storage unit, regarding the change in value of a field as at step **906**, the method **900** can be implemented, at step **908**, by checking by the listener whether to render a respective dynamic field, based on the change in the value of the field. Thereafter, at step **910**, the method includes determining by the listener whether or not to render the dynamic field or option. In case the dynamic field or option is to be rendered based on the rendering conditions and logical operator, the listener will send a rendering signal to the visual representation means. If, based on the input or change in value as detected at the storage unit, the corresponding rendering conditions and logical operator of the dynamic field or option are not satisfied, the listener may determine that dynamic field or option is not to be rendered.

[0081] **FIG. 10** is a flow diagram that illustrates an example method **920**, performed at a third-party system, for rendering a dynamic form, in accordance with the disclosed embodiments. As disclosed in **FIG. 10**, the example method **920** can be implemented, for example, at step **922**, by receiving, by a third-party system, the configuration of a dynamic form with fields, options and rendering conditions and logical operator. As disclosed earlier, the fields and options of a dynamic form are configured in a format that can be used for conveniently rendering at a third-party system. In some implementations of the present invention the format can be a JSON format. In some implementations of the present invention, the configuration of the dynamic form is received by the third-party using message queues or other methods such as API (Application Programming Interface) calls. Thereafter, the method **920** is implemented

at step **924**, by decoding by the third-party system, the dynamic form configuration and obtaining the dynamic form with fields and options and rendering conditions at step **926**. Finally, as at step **928**, the example method **920** is implemented by rendering the dynamic form on the GUI of the third-party system. In some implementations of the present invention, the dynamic form received at the third-party system may contain configurations for dynamic field with one or more rendering conditions and logical operators. However, such dynamic fields are rendered only when a value in a condition field changes and the rendering condition and logical operator in the dynamic field configuration is satisfied.

[0082] A computer readable medium is used for storing a program comprising instructions that, when executed by a system, cause the system to perform, the method for configuring and rendering a dynamic field and a dynamic option in a dynamic form. In one embodiment of the invention, the computing device of the visual representation means or GUI, may comprises the computer readable medium to perform the method. In another embodiment of the invention, the program in the computer readable medium can be the configuration program of the execution unit. The present invention enables the dynamic form to be easily customizable and rendering of the fields dynamically when certain conditions are satisfied, without having to change the underlying codes.

[0083] It will be appreciated that variations of the above disclosed and other features and functions, or alternatives thereof, may be desirably combined into many other different systems or applications. Also, various presently unforeseen or unanticipated alternatives, modifications, variations or improvements therein may be subsequently made by those skilled in the art which are also intended to be encompassed by the following claims.

[0084] Although embodiments of the current disclosure have been described comprehensively in considerable detail to cover the possible aspects, those skilled in the art would recognize that other versions of the disclosure are also possible.

CLAIMS

What is claimed is:

1. A system for configuring and rendering a dynamic field and a dynamic option in a dynamic form, the system comprising:
 - a visual representation means for visually displaying one or more of: a field, an option and a form, such that the visual representation means displays one or more of a fields configuration board or an options configuration board or a forms configuration board; and
 - an execution unit comprising at least one configuration program and at least one storage unit,
 - wherein the configuration program enables the configuration of the fields on the fields configuration board and the configuration of the options on the options configuration board;
 - wherein the configuration program enables the configuration of the dynamic field on the fields configuration board with at least one rendering condition and at least one logical operator;
 - wherein the dynamic field is rendered when the rendering condition is satisfied;
 - wherein the configuration program enables the configuration of the dynamic option on the options configuration board with at least one rendering condition and at least one logical operator;
 - wherein the dynamic option is rendered when the rendering condition is satisfied;
 - wherein the configuration program enables the configuration of the dynamic form on the forms configuration board with one or more of the dynamic field and the dynamic option; and
 - wherein the storage unit stores for each field, a field code and a value corresponding to the field.
2. The system of claim 1, wherein the rendering condition uses the field code and the value of the field.

3. The system of claim 1, wherein the dynamic field is configured and rendered on one or more page tabs.
4. The system of claim 1, wherein the field code and the value are stored in the storage unit when the field is configured.
5. The system of claim 1, wherein the dynamic field is rendered based on the rendering condition and the logical operator based on an other field.
6. The system of claim 5, wherein the other field is with the option.
7. The system of claim 1, wherein the fields configuration board for each field, enables defining one or more attributes, such that the attributes include a field type, a data type, a regex, a field code, a parent code, a maximum length or a description of the field.
8. The system of claim 1, wherein the options configuration board for each option enables defining one or more attributes, such that the attributes include a code value, a sequence or an options group code.
9. The system of claim 8, wherein the configuration program enables assigning the values to the attributes, defining the field code, setting the options and setting the rendering conditions and the logical operators to the options.
10. The system of claim 1, wherein the storage unit stores the dynamic form with the fields and the options configured using the configuration program.
11. The system of claim 1, wherein the dynamic form is synchronized and rendered to a third-party system.
12. The system of claim 1, wherein the fields configuration board is a fields configuration page.
13. The system of claim 1, wherein the options configuration board is an options

configuration page.

14. A method for configuring and rendering a dynamic field and a dynamic option in a dynamic form, the method comprising:
- configuring, by a fields configuration board in a visual representation means using a configuration program, a field in the dynamic form;
 - configuring, by an options configuration board in the visual representation means using the configuration program, an option in the dynamic form;
 - configuring, by the fields configuration board using the configuration program, the dynamic field in the dynamic form with at least one rendering condition and at least one logical operator; and
 - configuring, by the options configuration board using the configuration program, the dynamic option in the dynamic form with at least one rendering condition and at least one logical operator.
15. The method of claim 14, further comprising:
- initializing, by an execution unit, one or more fields, options and forms in an order of configuration;
 - registering, by the execution unit, each field in a storage unit with a field code and a value corresponding to the field along with the rendering condition and the logical operator;
 - checking, by the execution unit, the rendering condition and the logical operator of the dynamic field or the dynamic option to be rendered on other fields or options; and
 - determining, by the execution unit, renderability of the dynamic field based on the satisfaction of the rendering condition and the logical operator.
16. The method of claim 14, further comprising:
- receiving, by the execution unit, the value of one or more of the fields or the options in the dynamic form;
 - informing, by the execution unit, the value of the fields or the options to the storage unit;
 - detecting, by the execution unit, an input or change in the value of the fields or the options;

checking, by the execution unit, the rendering condition and the logical operator of the dynamic field or the dynamic option to be rendered on other fields or options; and

determining, by the execution unit, renderability of the dynamic field based on the satisfaction of the rendering condition and the logical operator.

17. The method of claim 15 and 16, further comprising:

sending, by the execution unit, an instruction to render the dynamic field or the dynamic option to the visual representation means; and

rendering, by the visual representation means, the dynamic field or the dynamic option in the dynamic form.

18. The method of claims 24, further comprising:

receiving, by a third-party system, the dynamic form with one or more of the dynamic field or the dynamic option;

decoding, by the third-party system, configuration of the dynamic form;

obtaining, by the third-party system, the dynamic form with one or more of the dynamic field or the dynamic option with the rendering condition and the logical operator; and

rendering, by the third-party system, the dynamic form.

19. A computer readable medium storing a program comprising instructions that, when executed by a system, cause the system to perform, the method of claims 14-18.

20. A computing device comprising the computer readable medium of claim 19.