Claims

1. A computer-implemented method for dynamically creating and integrating interactive web forms within a document, the method comprising:

    - converting a source document authored in a rich-text format into an HTML document comprising a plurality of HTML elements;

    - rendering the HTML document in a user interface such that the HTML elements are visually arranged according to the source document's layout;

    - detecting a drag-and-drop operation initiated by a user, the operation including a selection of a form input field from a palette and a drop target within the rendered HTML document, wherein the drop target is an HTML element selected from the group consisting of a table cell, a paragraph, a division, a span, a heading, or a list element;

    - in response to said detecting, programmatically embedding the selected form input field into the HTML document at the drop target location, thereby updating the HTML document to include an interactive form control without disrupting the document's structural integrity;

    - linking the embedded form input field to a data storage system by associating the form input field with a database field or data object, such that any user-provided input to the form input field is transmitted and stored in the data storage system in real time or upon form submission;

    - and upon receiving user input in the embedded form input field, updating the data storage system with the input while preserving an association between the input, the document, and the user who provided it, whereby the document is transformed into an interactive "living" form that captures data natively in HTML and synchronizes with back-end databases.

2. The method of claim 1, further comprising recognizing a plurality of different HTML element types as valid drop targets for form input fields by monitoring drag-over events, and providing a visual highlight or indicator on an HTML element when it is eligible to receive a dropped form input field, thereby guiding the user in where form fields can be placed within the document .

3. The method of claim 1, wherein embedding the form input field includes adjusting the display style of the form input field based on the tag of the drop target element – such that if the target element is an inline element or part of flowing text, the form input field is styled as inline or inline-block, and if the target is a block-level container, the form input field is styled as block – thereby ensuring the inserted field conforms to the document's

format .

4. The method of claim 1, further comprising generating an audit log entry each time a form input field is modified or a form is submitted, the audit log entry including at least a timestamp, the identity of the user, and an identifier of the form input field and document, such that a secure chronological record of user actions on the interactive document is maintained .

5. The method of claim 1, wherein the source document is selected from the group consisting of: a Microsoft Word document, a Google Docs document, a LibreOffice document, or a Markdown text file, and wherein converting the source document into HTML includes parsing the source format and generating semantically equivalent HTML elements, including tables, lists, and text styling, so that the converted HTML document retains the content and structure of the source document in a web-compatible form.

6. The method of claim 1, further comprising deploying the HTML document with embedded form input fields to a client device and enabling offline operation by: caching the HTML document and associated script assets on the client device; storing any user inputs to form fields in a local memory of the client device when the device is offline; and upon detection of network connectivity, automatically synchronizing the stored inputs with the remote data storage system, thereby ensuring form data is captured during offline periods and later merged with central databases without loss .

7. The method of claim 6, wherein the client device is a portable or edge device selected from the group consisting of a rugged tablet, a smartphone, a wearable AR display, or a laptop, and wherein the method further includes authenticating the user on the client device via a biometric sensor (fingerprint scanner, facial recognition, or retinal scanner) prior to allowing access to or submission of the interactive document, such that the identity of the user can be verified in compliance with electronic signature regulations .

8. The method of claim 1, further comprising: defining one or more event-driven triggers in a trigger management module, each trigger associated with a condition (selected from a sensor input event, a message or signal from an external system, a specific time or interval, or a predefined state change in the data storage system) and linked to a predetermined HTML template or interactive document; monitoring for occurrence of the condition; and automatically instantiating or displaying the predetermined interactive document with embedded form fields to a target user or user group upon detection of the condition, thereby launching a relevant workflow or form in real-time response to external events without manual initiation.

9. The method of claim 8, wherein the event-driven trigger is implemented via an event streaming platform or message queue, and the condition comprises receiving a published message on a topic that matches criteria defined for the trigger (including content of the message or type of event), and wherein automatically instantiating the

interactive document includes transmitting a notification or instruction to a user device to open the corresponding interactive HTML form template for the user's immediate action.

10. The method of claim 8, wherein at least one trigger condition is a time-based rule that causes the system to generate and assign a recurring form (with a specified document template) at set intervals or deadlines, thereby ensuring periodic tasks or checklists are presented to users at scheduled times (for example, a daily safety checklist that appears each morning for a facility manager).

11. The method of claim 1, further comprising providing an administrative configuration interface that allows authorized users to: upload or import source documents for conversion; graphically add or remove form fields on the HTML document; specify data bindings between form fields and fields in enterprise databases; define validation rules for inputs (including data type checks, ranges, or mandatory fields); and configure user access permissions for the interactive document or specific fields, wherein changes made via this interface are applied without requiring manual coding, thereby enabling codeless administration of interactive forms.

12. A system for integrating live forms into documents, comprising:

    - one or more processors and a memory coupled thereto;

    - a document conversion module stored in the memory and configured to, when executed, receive an input document in a first format and output a corresponding HTML document with structural elements mirroring the input document's content;

    - a user interface module configured to display the HTML document to a user on a client device and to handle user interactions, including drag-and-drop operations and form field entries;

    - a form embedding engine configured to create interactive form elements in real time within the HTML document in response to user drag-and-drop input, the form embedding engine further ensuring that inserted form elements are tagged and styled appropriately within the HTML structure;

    - a data integration module configured to link each interactive form element to a back-end data repository, the data integration module capable of sending user-entered data from the client device to the data repository and retrieving data from the repository to pre-fill form elements, thereby synchronizing the document's form content with enterprise data;

    - a workflow trigger module configured to detect specified external or internal events and, upon detection, automatically instantiate an interactive document

template for user interaction as per the detected event;

- and an audit log database or file operatively connected to the data integration module, wherein the system records key events including form field modifications, submissions, and user authentications to the audit log in a manner that is secure and tamper-evident.

13. The system of claim 12, wherein the user interface module is further configured to operate in a web browser environment on the client device and utilize standard web technologies (HTML5, CSS, JavaScript) such that the interactive document is accessible on any device with a web browser, and wherein the system supports a progressive web app mode for offline use, storing necessary files in a browser cache and utilizing local storage or an embedded database on the client device for interim data when offline.

14. The system of claim 12, wherein the client device includes at least one of: a touchscreen for direct manipulation of the document content (to drag fields into place or fill them), a camera for scanning codes or taking images to attach via the form, and sensors or external device interfaces (such as a barcode scanner or RFID reader) that the system can interface with to input data into form fields automatically, thereby streamlining data entry in field operations.

15. The system of claim 12, wherein the workflow trigger module comprises a subscription to an event bus and the system further includes a notification subsystem that can push alerts to users (via mobile push notifications, emails, or on-screen alerts) informing them of a newly instantiated form or required action, such that event-driven forms gain immediate user attention and quick response.

16. The system of claim 12, wherein the data integration module includes connectors for multiple types of enterprise software systems, selected from: ERP systems, Quality Management Systems, CRM systems, Learning Management Systems, and others, allowing the system to both fetch reference data to present in forms (e.g., pulling a list of approved materials into a dropdown) and to post completed form data into these systems (e.g., recording a quality issue in a QMS when a form indicating a deviation is submitted).

17. The system of claim 12, wherein the audit log stores entries with a digital signature or hash chaining such that each entry is cryptographically linked to the previous one (for example, using a blockchain ledger or hash chain) to prevent undetected alteration, and wherein the system provides an interface for authorized auditors to review the audit trail filtered by document, user, or date, facilitating regulatory compliance reviews.

18. The system of claim 12, further comprising a deployment configuration that allows the system to be run in a cloud environment with compliance certifications, including a mode for deployment to a government cloud meeting FedRAMP High and DoD Impact Level 5

requirements, wherein all data is encrypted and the system is isolated to US persons, thereby enabling use of the system for sensitive government workflows .

19. An interactive document system as in claim 12, wherein the interactive document templates are version-controlled, such that any change to the content or embedded fields of a template is recorded as a new version, and active instances of forms are linked to the version they were created from, thereby enabling traceability of which procedural version was followed for each data submission and simplifying compliance with document control standards (including providing the capability to recall or invalidate obsolete versions enterprise-wide when a new version is approved).

20. A non-transitory computer-readable medium storing instructions which, when executed by one or more processors, cause a system to perform the method according to any of claims 1 through 11.

(Each of the above claims should be considered as standing on its own as separately patentable, and combinations of features from dependent claims in different permutations may be pursued in further claim sets as the patent prosecution may demand. The invention is not limited by the example implementations or specific technologies mentioned, but rather defined by the claims and their equivalents.)