CLAIMS

What is claimed is:

1. A computer implemented method for generating and deploying a code block in a blockchain, the method comprising:

rendering, by a network node, a graphical user interface containing a plurality of contract components and a plurality of document components retrieved from a library database of contract and document components, wherein the plurality of document components comprise descriptive text;

generating, by the network node, an assembled contract text based on a selection of at least one contract component from the plurality of contract components or at least one document component from the plurality of document components;

retrieving, by the network node, an address of an executable code corresponding to the at least one contract component or an address of a text of the at least one document component from the library database of contract and document components;

retrieving, by the network node, a first latest valid blockchain based upon a first polling of a plurality of network nodes using a consensus threshold;

generating, by the network node, a code block containing the address of the executable code corresponding to the at least one contract component or the address of the text of the at least one document component;

generating, by the network node, a block address for the code block based upon a cryptographic hash value of at least a portion of data records of the last block of the first latest valid blockchain; and

appending, by the network node, the code block at the generated block address in the first latest valid blockchain.

2. The method of claim 1, further comprising

in response to detecting a digital event trigger by the network node:

associating, by the network node, the digital event trigger to the deployed code block;

retrieving, by the network node, a second latest valid blockchain based upon a second polling of a plurality of network nodes using the consensus threshold;

downloading, by the network node, the code block to the memory of the network node;

retrieving, by the network node, an executable code corresponding to the at least one contract component based on the address of the executable code; and

executing, by the network node, the executable code to generate an output.

3. The method of claim 2, further comprising:

generating, by the network node, a blockchain block containing the output; and

appending, by the network node, the blockchain block to the second latest valid blockchain.

4. The method of claim 1, wherein the graphic user interface

renders the at least one contract component from the plurality of contract components or the at least one document component from the plurality of document components as an icon, a dialog box, or an option that a user can drag and drop on a contract template.

5. The method of claim 2, wherein the address of the executable code is a block address within each of the first and second latest valid blockchains; and/or wherein the address of the executable code is an address of a database record in the library database of contract and document components.

6. The method according to any one of claims 1 to 5, wherein the blockchain is a private party blockchain.

7. The method according to any one of claims 1 to 6, wherein the graphical user interface is a what-you-see-is-what-you-get, WYSIWYG, graphical user interface.

8. A computer implemented method for generating and deploying a code block in a blockchain, the method comprising:

rendering, by a network node, a graphical user interface containing a plurality of contract components and a plurality of document components retrieved from a library database of contract and document components, wherein the plurality of document components comprise descriptive text;

generating, by the network node, an assembled contract text based on a selection of at least one contract component from the plurality of contract components or at least one document component from the plurality of document components;

retrieving, by the network node, an executable code corresponding to the at least one contract component or a text of the at least one document component from the library database of contract and document components;

retrieving, by the network node, a first latest valid blockchain based upon a first polling of a plurality of network nodes using a consensus threshold;

generating, by the network node, a code block containing the executable code corresponding to the at least one contract component or the text of the at least one document component;

generating, by the network node, a block address for the code block based upon a cryptographic hash value of at least a portion of data records of the last block of the first latest valid blockchain; and

appending, by the network node, the code block at the generated block address in the first latest valid blockchain.

9. The method of claim 8, further comprising

in response to detecting a digital event trigger by the network node:

associating, by the network node, the digital event trigger to the deployed code block;

retrieving, by the network node, a second latest valid blockchain based upon a second polling of a plurality of network nodes using the consensus threshold;

downloading, by the network node, the code block to the memory of the network node; and

executing, by the network node, the executable code in the code block to generate an output.

10. The method of claim 9, further comprising:

generating, by the network node, an output block containing the output; and

appending, by the network node, the output block to the second latest valid blockchain.

11. The method of claims 2 or 9, wherein the digital event trigger comprises at least one of: a predetermined date, a predetermined point in time, a predetermined duration of time; a predetermined number in a counter, an outcome of an intelligent autocalculation, an outcome of an API event, an outcome of a blockchain event, a user action on a web interface, and an input from an admin user.

12. The method of claim 8, further comprising:

receiving, by the network node, a contract component or a document component from a user;

generating, by the network node, the assembled contract text to include the contract component or the document component received from the user; and

updating, by the network node, the library database to include the contract component or the document component received from the user.

13. The method according to any one of claims 8 to 12, wherein the blockchain is a private party blockchain.

14. The method according to any one of claims 8 to 13, wherein the graphical user interface is a what-you-see-is-what-you-get, WYSIWYG, graphical user interface.

15. A system for generating and deploying a code block in a blockchain, the system comprising:

a plurality of network nodes, each of the network nodes including a non-transitory storage medium storing a respective local copy of a blockchain;

at least one of the plurality of network nodes having a processor configured to:

render a graphical user interface containing a plurality of contract components and a plurality of document components retrieved from a library database of contract and document components, wherein the plurality of document components comprise descriptive text;

generate an assembled contract text based on a selection of at least one contract component from the plurality of contract components or at least one document component from the plurality of document components;

retrieve an executable code, corresponding to the at least one contract component or a text of the at least one document component from the library database of contract and document components;

retrieve a first latest valid blockchain based upon a first polling of a plurality of network nodes using a consensus threshold;

generate a code block containing the executable code corresponding to the at least one contract component or the text of the at least one document component;

generate a block address for the code block based upon a cryptographic hash value of at least a portion of data records of the last block of the first latest valid blockchain; and

append the code block at the generated block address in the first latest valid blockchain.

16. The system of claim 15, wherein the processor is further configured to:

in response to detecting a digital event trigger:

associate the digital event trigger to the code block;

retrieve the latest valid blockchain based upon a second polling of a plurality of network nodes using the consensus threshold;

download the code block to the memory of the network node;

execute the executable code in the code block to generate an output.

17. The system of claim 16, wherein the processor is further configured to:

generate an output block containing the output; and

append the output block to the second latest valid blockchain.

18. The system of claim 16, wherein the digital event trigger comprises at least one of: a predetermined date, a predetermined point in time, a predetermined duration of time; a predetermined number in a counter, an outcome of an intelligent autocalculation, an outcome of an API event, an outcome of a blockchain event, a user action on a web interface, and an input from an admin user.

19. The system of claim 15, wherein the graphical user interface

renders the at least one contract component from the plurality of contract components or the at least one document component from the plurality of document components an icon, a dialog box, or an option that a user can drag and drop on a contract template.

20. The system according to any one of claims 15 to 19, wherein the blockchain is a private party blockchain.

21. The system according to any one of claims 15 to 20, wherein the graphical user interface is a what-you-see-is-what-you-get, WYSIWYG, graphical user interface.