

INVOICE INVALIDATION METHOD AND APPARATUS BASED ON BLOCKCHAIN, AND ELECTRONIC DEVICE

TECHNICAL FIELD

[0001] One or more implementations of the present specification relate to the field
5 of blockchain technologies, and in particular, to an invoice invalidation method and
apparatus based on a blockchain, and an electronic device.

BACKGROUND

[0002] A blockchain technology, also known as a distributed ledger technology, is
an emerging technology in which several computing devices participate in
10 "bookkeeping" and jointly maintain a complete distributed database. Because the
blockchain technology is characterized by being decentralized, open and transparent,
enabling each computing device to participate in database recording, and enabling
data synchronization to be performed quickly between the computing devices, the
blockchain technology is used to build a decentralized system, and various executable
15 programs are stored in the distributed database of the blockchain for automatic
execution, which has been widely used in many fields.

SUMMARY

[0003] The present specification provides an invoice invalidation method based on
a blockchain, including: receiving an invalidation request that is initiated by an
20 invoice invalidation party for a target invoice recorded in the blockchain; in response
to the invalidation request, determining whether the invoice invalidation party has
authority to invalidate the target invoice; and in response to that the invoice
invalidation party has the authority to invalidate the target invoice, invoking an
invoice invalidation logic provided in a smart contract deployed in a blockchain
25 network of the blockchain to invalidate the target invoice.

[0004] Optionally, the smart contract further provides authority verification logic
for the invoice invalidation party; and the determining whether the invoice
invalidation party has authority to invalidate the target invoice includes: invoking the

authority verification logic provided in the smart contract to determine whether the invoice invalidation party has authority to invalidate the target invoice.

[0005] Optionally, the invalidation request includes a user identifier of the invoice invalidation party; and the determining whether the invoice invalidation party has authority to invalidate the target invoice includes: determining, based on a user identifier, whether the invoice invalidation party is an issuer of the target invoice or a trusted third party; and in response to that the invoice invalidation party is an issuer of the target invoice or a trusted third party, determining that the invoice invalidation party has the authority to invalidate the target invoice.

[0006] Optionally, the invalidation request includes an invoice identifier of the target invoice; and the method further includes: determining, based on an invoice identifier, whether an invoice corresponding to the invoice identifier is stored in the blockchain; and in response to that the invoice corresponding to the invoice identifier is stored in the blockchain, determining the invoice as the target invoice.

[0007] Optionally, the invoice identifier includes: a hash value obtained through hash calculation based on an invoice content; or a hash value obtained through hash calculation based on a piece of uniqueness information in the invoice content.

[0008] Optionally, the invalidating the target invoice includes: updating a state of the target invoice from a valid state to an invalid state.

[0009] The present specification further provides an invoice invalidation method based on a blockchain, including: obtaining a transaction that has been recorded in the blockchain; determining whether the transaction is a refund transaction corresponding to a payment transaction that has been recorded in the blockchain; and in response to that the transaction is a refund transaction corresponding to the payment transaction, invoking an invoice invalidation logic provided in a smart contract deployed in a blockchain network of the blockchain to invalidate a target invoice that is related to the payment transaction and recorded in the blockchain.

[0010] Optionally, the smart contract further provides transaction matching logic; and the determining whether the transaction is a refund transaction corresponding to the payment transaction that has been recorded in the blockchain includes: invoking the transaction matching logic provided in the smart contract to determine whether the transaction is a refund transaction corresponding to the payment transaction that has been recorded in the blockchain.

[0011] Optionally, the method further includes: determining, based on an invoice identifier corresponding to the payment transaction, whether an invoice corresponding to the invoice identifier is stored in the blockchain; and in response to that the invoice corresponding to the invoice identifier is stored in the blockchain, determining the
5 invoice as the target invoice.

[0012] Optionally, the invoice identifier includes: a hash value obtained through hash calculation based on an invoice content; or a hash value obtained through hash calculation based on a piece of uniqueness information in the invoice content.

[0013] Optionally, the invalidating the target invoice that is related to the payment
10 transaction and recorded in the blockchain includes: updating a state of the target invoice that is related to the payment transaction and recorded in the blockchain from a valid state to an invalid state.

[0014] The present specification further provides an invoice invalidation apparatus based on a blockchain, including: a receiving module, configured to receive an
15 invalidation request that is initiated by an invoice invalidation party for a target invoice recorded in the blockchain; a first determining module, configured to: in response to the invalidation request, determine whether the invoice invalidation party has authority to invalidate the target invoice; and an invalidation module, configured to: in response to that the invoice invalidation party has the authority to invalidate the
20 target invoice, invoke an invoice invalidation logic provided in a smart contract deployed in a blockchain network of the blockchain to invalidate the target invoice.

[0015] Optionally, the smart contract further provides authority verification logic for the invoice invalidation party; and the first determining module is specifically configured to: invoke the authority verification logic provided in the smart contract to
25 determine whether the invoice invalidation party has authority to invalidate the target invoice.

[0016] Optionally, the invalidation request includes a user identifier of the invoice invalidation party; and the first determining module is specifically configured to: determine, based on the user identifier, whether the invoice invalidation party is an
30 issuer of the target invoice or a trusted third party; and in response to that the invoice invalidation party is an issuer of the target invoice or a trusted third party, determine that the invoice invalidation party has the authority to invalidate the target invoice.

[0017] Optionally, the invalidation request includes an invoice identifier of the target invoice; and the apparatus further includes: a second determining module,

configured to: determine, based on an invoice identifier, whether an invoice corresponding to the invoice identifier is stored in the blockchain; and a third determining module, configured to: when the invoice corresponding to the invoice identifier is stored in the blockchain, determine the invoice as the target invoice.

5 **[0018]** Optionally, the invoice identifier includes: a hash value obtained through hash calculation based on an invoice content; or a hash value obtained through hash calculation based on a piece of uniqueness information in the invoice content.

[0019] Optionally, the invalidation module is specifically configured to: update a state of the target invoice from a valid state to an invalid state.

10 **[0020]** The present specification further provides an invoice invalidation apparatus based on a blockchain, including: an acquisition module, configured to obtain a transaction that has been recorded in the blockchain; a first determining module, configured to determine whether the transaction is a refund transaction corresponding to a payment transaction that has been recorded in the blockchain; and an invalidation
15 module, configured to: when the transaction is a refund transaction corresponding to the payment transaction, invoke an invoice invalidation logic provided in a smart contract deployed in a blockchain network of the blockchain to invalidate a target invoice that is related to the payment transaction and recorded in the blockchain.

[0021] Optionally, the smart contract further provides transaction matching logic;
20 and the first determining module is specifically configured to: invoke the transaction matching logic provided in the smart contract to determine whether the transaction is a refund transaction corresponding to the payment transaction that has been recorded in the blockchain.

[0022] Optionally, the apparatus further includes: a second determining module,
25 configured to: determine, based on an invoice identifier corresponding to the payment transaction, whether an invoice corresponding to the invoice identifier is stored in the blockchain; and a third determining module, configured to: when the invoice corresponding to the invoice identifier is stored in the blockchain, determine the invoice as the target invoice.

30 **[0023]** Optionally, the invoice identifier includes: a hash value obtained through hash calculation based on an invoice content; or a hash value obtained through hash calculation based on a piece of uniqueness information in the invoice content.

[0024] Optionally, the invalidation module is specifically configured to: update a state of the target invoice that is related to the payment transaction and recorded in the blockchain from a valid state to an invalid state.

[0025] The present specification further provides an electronic device, including: a processor; and a memory configured to store machine-executable instructions; where
5 by reading and executing the machine-executable instructions stored in the memory and corresponding to control logic for invoice invalidation based on a blockchain, the processor is enabled to: receive an invalidation request that is initiated by an invoice invalidation party for a target invoice recorded in the blockchain; in response to the
10 invalidation request, determine whether the invoice invalidation party has authority to invalidate the target invoice; and in response to that the invoice invalidation party has the authority to invalidate the target invoice, invoke an invoice invalidation logic provided in a smart contract deployed in a blockchain network of the blockchain to invalidate the target invoice.

[0026] The present specification further provides an electronic device, including: a processor; and a memory configured to store machine-executable instructions; where
15 by reading and executing the machine-executable instructions stored in the memory and corresponding to control logic for invoice invalidation based on a blockchain, the processor is enabled to: obtain a transaction that has been recorded in the blockchain;
20 determine whether the transaction is a refund transaction corresponding to a payment transaction that has been recorded in the blockchain; and in response to that the transaction is a refund transaction corresponding to the payment transaction, invoke an invoice invalidation logic provided in a smart contract deployed in a blockchain network of the blockchain to invalidate a target invoice that is related to the payment
25 transaction and recorded in the blockchain.

[0027] In the above technical solutions, when an invoice invalidation party initiates an invalidation request on a certain invoice recorded in a blockchain, corresponding logic in a smart contract deployed in a blockchain network of the blockchain is invoked to verify invalidation authority of the invoice invalidation party, and when it
30 is determined that the invoice invalidation party has the authority to invalidate the invoice, the invoice is invalidated. Alternatively, when it is determined that a certain transaction that has been recorded in a blockchain is a refund transaction corresponding to a payment transaction that has been recorded in the blockchain, corresponding logic in a smart contract deployed in a blockchain network of the

blockchain is invoked to invalidate an invoice that is related to the payment transaction and recorded in the blockchain. As such, the invoice recorded in the blockchain is invalidated, and the data security of invoices circulating in the blockchain is ensured.

5

BRIEF DESCRIPTION OF DRAWINGS

[0028] FIG. 1 is a flowchart illustrating an invoice invalidation method based on a blockchain according to some example implementations of the present specification;

[0029] FIG. 2 is a flowchart illustrating another invoice invalidation method based on a blockchain according to some example implementations of the present specification;

10

[0030] FIG. 3 is a diagram illustrating a hardware structure of an electronic device where an invoice invalidation apparatus based on a blockchain is located according to some example implementations of the present specification;

[0031] FIG. 4 is a block diagram illustrating an invoice invalidation apparatus based on a blockchain according to some example implementations of the present specification; and

15

[0032] FIG. 5 is a block diagram illustrating another invoice invalidation apparatus based on a blockchain according to some example implementations of the present specification.

20

DESCRIPTION OF IMPLEMENTATIONS

[0033] Example implementations are described in detail herein, and examples thereof are shown in the accompanying drawings. When the following description refers to the accompanying drawings, unless otherwise indicated, the same numbers in different accompanying drawings refer to the same or similar elements. The following example implementations do not represent all implementations consistent with one or more implementations of the present specification. In contrast, they are merely examples of apparatuses and methods consistent with some aspects of one or more implementations of the present specification as described in detail in the appended claims.

25

[0034] Terms used in the present specification are for the purpose of describing specific implementations only and are not intended to limit the present specification. As used in the present specification and the appended claims, the singular forms "a (an)", "the" and "this" are also intended to include the plural forms, unless the context
5 clearly indicates other meaning. It should also be understood that the term "and/or" as used herein refers to and includes any or all possible combinations of one or more associated listed items.

[0035] It should be understood that although the terms such as "first", "second", and "third" can be used to describe various types of information in the present
10 specification, the information should not be limited to these terms. These terms are only used to differentiate information of a same type. For example, without departing from the scope of the present specification, the first information can also be called second information, and similarly, the second information can also be called first information. Depending on the context, the word "if" as used herein can be explained
15 as "when" or "in response to determining".

[0036] The present specification provides technical solutions for invalidating an invoice recorded in a blockchain.

[0037] In some implementations, when a user needs to invalidate a certain invoice recorded in a blockchain of a blockchain network, the user (called invoice invalidation
20 party) can initiate an invalidation request on the invoice (called target invoice) by using a client.

[0038] After receiving the invalidation request, a node device in the blockchain network can respond to the invalidation request.

[0039] Specifically, it can be determined first whether the invoice invalidation
25 party has authority to invalidate the target invoice. Generally, only an invoice issuer or a trusted third party (such as a tax bureau or an industrial and commercial bureau) is allowed to invalidate a certain invoice, so as to ensure the data security of the invoice and prevent the invoice from being maliciously invalidated. In this case, it can be first determined whether the invoice invalidation party that initiated the invalidation
30 request has authority to invalidate the target invoice. In response to that the invoice invalidation party is an issuer of the invoice or a trusted third party, it can be determined that the invoice invalidation party has the authority to invalidate the invoice.

[0040] After it is determined that the invoice invalidation party has the authority to invalidate the invoice, the node device can invoke an invoice invalidation logic provided in a smart contract deployed, *e.g.*, in the blockchain network to invalidate the target invoice recorded in the blockchain.

5 **[0041]** In some other implementations, after completing a payment or refund, a user can record the payment transaction or refund transaction in a blockchain. A node device in the blockchain network can obtain a transaction that has been recorded in the blockchain, and determine whether the transaction is a refund transaction corresponding to a certain payment transaction that has been recorded in the
10 blockchain.

[0042] For example, it can be checked whether transaction objects and transaction content (such as transaction amount and transaction items) in the transaction match the transaction objects and transaction content of each payment transaction that has been recorded in the blockchain. In response to that the payer of the transaction is a
15 payee of another payment transaction that has been recorded in the blockchain, the payee of the transaction is a payer of the other payment transaction, and the transaction content of the transaction is the same as that of the other payment transaction, it can be determined that the transaction is a refund transaction corresponding to the payment transaction.

20 **[0043]** After it is determined that the transaction is a refund transaction corresponding to a certain payment transaction that has been recorded in the blockchain, a node device can invoke an invoice invalidation logic provided in a smart contract deployed in the blockchain network to invalidate a target invoice that is related to the payment transaction and recorded in the blockchain.

25 **[0044]** In the above technical solutions, when an invoice invalidation party initiates an invalidation request on a certain invoice recorded in a blockchain, corresponding logic in a smart contract deployed in a blockchain network of the blockchain is invoked to verify invalidation authority of the invoice invalidation party, and when it is determined that the invoice invalidation party has the authority to invalidate the
30 invoice, the invoice is invalidated. Alternatively, when it is determined that a certain transaction that has been recorded in a blockchain is a refund transaction corresponding to a payment transaction that has been recorded in the blockchain, corresponding logic in a smart contract deployed in a blockchain network of the blockchain is invoked to invalidate an invoice that is related to the payment

transaction and recorded in the blockchain. As such, the invoice recorded in the blockchain is invalidated, and the data security of invoices circulating in the blockchain is ensured.

5 [0045] The present specification is further described below with reference to specific implementations.

[0046] FIG. 1 is a flowchart illustrating an invoice invalidation method based on a blockchain according to some example implementations of the present specification. The method can be applied to a node device in a blockchain network, and includes the following steps.

10 [0047] Step 102: Receive an invalidation request that is initiated by an invoice invalidation party for a target invoice recorded in the blockchain.

[0048] Step 104: In response to the invalidation request, determine whether the invoice invalidation party has authority to invalidate the target invoice.

15 [0049] Step 106: in response to that the invoice invalidation party has the authority to invalidate the target invoice, invoke an invoice invalidation logic provided in a smart contract deployed in a blockchain network of the blockchain to invalidate the target invoice.

20 [0050] The blockchain network described in the present specification can specifically include any type of blockchain network. For example, in practice, any one of a public blockchain, a private blockchain and a consortium blockchain network can be used.

[0051] Because data stored in a blockchain cannot be tampered with, it has become the mainstream trend in the industry to store the data in the blockchain for data record, *i.e.*, to store data as record in the blockchain.

25 [0052] In the implementations, the invoice can be recorded in the blockchain.

[0053] When an invoice invalidation party needs to invalidate a target invoice recorded in the blockchain, the invoice invalidation party can initiate, by using a client, a transaction for invalidating the target invoice, *e.g.*, initiate an invalidation request on the target invoice by using the client.

30 [0054] Transactions in the blockchain are divided into transactions in narrow meanings and transactions in broad meanings. In narrow meanings, a transaction refers to a value transfer deployed by a user to the blockchain. For example, in a conventional bitcoin blockchain network, a transaction can be a transfer initiated by a user in the blockchain. In broad meanings, a transaction refers to a piece of service or

business data with a service or business intention published by a user to the blockchain. For example, the operator can build a consortium blockchain based on actual service requirements, and deploy, based on the consortium blockchain, some other online services unrelated to value transfer (such as a rental service, a vehicle
5 dispatching service, an insurance claims service, a credit service, and a medical service). In this consortium blockchain, the transaction can be a service/business message or service request with a service/business intention published by the user in the consortium blockchain.

[0055] After receiving the invalidation request, a node device in the blockchain can
10 respond to the invalidation request.

[0056] Specifically, it can be determined first whether the invoice invalidation party has authority to invalidate the target invoice.

[0057] If it is determined that the invoice invalidation party has the authority to invalidate the target invoice, invoice invalidation logic provided in a smart contract
15 deployed in a blockchain network of the blockchain can be invoked to invalidate the target invoice.

[0058] The invoice invalidation logic can be program code (such as some program methods or functions that can be invoked) that is provided in the smart contract and related to execution logic for invalidating an invoice.

[0059] In practice, the above smart contract can further provide authority verification logic for the above invoice invalidation party. In this case, the authority verification logic provided in the smart contract can be invoked to determine whether the invoice invalidation party has the above authority to invalidate the target invoice.

[0060] The authority verification logic can be specifically program code that is
25 provided in the smart contract and related to execution logic for verifying invalidation authority of an invoice invalidation party.

[0061] It should be noted that the smart contract for invoice invalidation described above and the smart contract for verifying invalidation authority of an invoice invalidation party can be integrated into one smart contract and deployed in the above
30 blockchain, or can be deployed in the blockchain as two different smart contracts, which is not limited in the present specification.

[0062] In some illustrated implementations, the invalidation request that is initiated by the invoice invalidation party for the target invoice by using the client can include a user identifier of the invoice invalidation party, so that it can be determined, based

on the user identifier of the invoice invalidation party, whether the invoice invalidation party has the authority to invalidate the target invoice.

[0063] For example, by checking whether the user identifier of the invoice invalidation party matches a user identifier of an issuer of the target invoice, it can be determined whether the invoice invalidation party is the issuer of the target invoice. In response to that the user identifier of the invoice invalidation party is the same as the user identifier of the issuer of the target invoice, it can be determined that the invoice invalidation party is the issuer of the target invoice, and thus it can be determined that the invalidation party has the authority to invalidate the target invoice.

[0064] For example, by checking whether the user identifier of the invoice invalidation party matches a user identifier of a trusted third party, it can be determined whether the invoice invalidation party is the trusted third party. In response to that the user identifier of the invoice invalidation party is the same as the user identifier of the trusted third party, it can be determined that the invoice invalidation party is the trusted third party, and thus it can be determined that the invoice invalidation party has the authority to invalidate the target invoice.

[0065] Accordingly, in response to that it is determined that the invoice invalidation party is neither the issuer of the target invoice nor the trusted third party, it can be determined that the invoice invalidation party does not have the authority to invalidate the target invoice. In this case, the target invoice cannot be subsequently invalidated.

[0066] The user identifier can be a taxpayer identification number, or an institution code of a trusted third party such as a tax bureau or an industrial and commercial bureau; alternatively, the user identifier can be a user number assigned to an individual, an enterprise or a tax bureau, and user numbers assigned to different users are different, which is not limited in the present specification.

[0067] For example, the user identifier can be a taxpayer identification number. After the invalidation request that is initiated for the target invoice and includes the user identifier of the invoice invalidation party is received, the taxpayer identification number of the invoice issuer can be obtained from the target invoice first, and then it is checked whether the obtained taxpayer identification number of the invoice issuer matches the user identifier of the invoice invalidation party. In response to that the taxpayer identification number of the invoice issuer is same as the user identifier of the invoice invalidation party, it can be determined that the invoice invalidation party

is the issuer of the target invoice, and thus it can be determined that the invalidation party has the authority to invalidate the target invoice.

[0068] For example, the user identifier can be an institution code of a trusted third party such as a tax bureau or an industrial and commercial bureau. After the
5 invalidation request that is initiated for the target invoice and includes the user identifier of the invoice invalidation party is received, it can be checked whether the institution code of each trusted third party matches the user identifier of the invoice invalidation party. In response to that the user identifier of the invoice invalidation party is the same as the institution code of a certain trusted third party, it can be
10 determined that the invoice invalidation party is the trusted third party, and thus it can be determined that the invoice invalidation party has the authority to invalidate the target invoice.

[0069] In another example, when an invoice is recorded in a blockchain, as, *e.g.*, described above, the user identifier of the invoice issuer or the trusted third party with
15 the authority to invalidate the invoice can be correspondingly recorded in the blockchain.

[0070] After the invalidation request that is initiated for the target invoice and includes the user identifier of the invoice invalidation party is received, it can be checked whether the user identifier of the invoice invalidation party matches a user
20 identifier corresponding to the target invoice. In response to that the user identifier of the invoice invalidation party is the same as a certain user identifier corresponding to the target invoice, it can be determined that the invoice invalidation party has the authority to invalidate the target invoice.

[0071] For example, assuming that a user identifier of a certain trusted third party
25 is user identifier 1, a user identifier of an issuer of invoice 1 is user identifier 2, and a user identifier of an issuer of invoice 2 is user identifier 2, then the invoice and the user identifier of the invoice issuer or trusted third party with the authority to invalidate the invoice can be recorded in the blockchain in the form shown in Table 1.

Table 1

Invoice	User identifier
Invoice 1	User identifier 1
	User identifier 2
Invoice 2	User identifier 1

	User identifier 3
...	...

[0072] After an invalidation request that is initiated for invoice 1 and includes user identifier 1 is received, it can be determined, based on Table 1, that the invoice invalidation party initiating the invalidation request has the authority to invalidate invoice 1; after an invalidation request that is initiated for invoice 1 and includes user identifier 3 is received, it can be determined, based on Table 1, that the invoice invalidation party initiating the invalidation request does not have the authority to invalidate invoice 1; and so on.

[0073] In some illustrated implementations, the invalidation request that is initiated by the invoice invalidation party for the target invoice by using the client can include an invoice identifier of the target invoice, so that based on the invoice identifier of the target invoice, it can be first determined whether the target invoice has been recorded in the blockchain, and when it is determined that the target invoice has been recorded in the blockchain, it can be determined whether the invoice invalidation party has the authority to invalidate the target invoice, so as to avoid invalid operation when the target invoice has not been recorded in the blockchain, and reduce device resource consumption.

[0074] For example, when an invoice is recorded in the blockchain, the invoice identifier of the invoice can be correspondingly recorded in the blockchain.

[0075] After the invalidation request that is initiated for the target invoice and includes the invoice identifier of the target invoice is received, it can be checked whether the invoice identifier of the target invoice matches an invoice identifier of an invoice that has been recorded in the blockchain. In response to that the invoice identifier of the invoice is the same as an invoice identifier of a certain invoice that has been recorded in the blockchain, it can be determined that the blockchain stores an invoice corresponding to the invoice identifier, and the invoice is determined as the target invoice. Subsequently, it can be determined whether the invoice invalidation party has authority to invalidate the target invoice.

[0076] Accordingly, in response to that it is determined that the blockchain does not store an invoice corresponding to the invoice identifier, it is no longer necessary to determine whether the invoice invalidation party has the authority to invalidate the target invoice.

[0077] The invoice identifier can be a hash value obtained through hash calculation based on an invoice content; or a hash value obtained through hash calculation based on a piece of uniqueness information (such as an invoice number and an invoice code) in the invoice content.

5 **[0078]** For example, assuming that an invoice identifier of invoice 1 is invoice identifier 1 and an invoice identifier of invoice 2 is invoice identifier 2, the invoice 1, the invoice identifier 1, the invoice 2, and the invoice identifier 2 can be recorded in the blockchain in the form shown in Table 2.

Table 2

Invoice	Invoice identifier
Invoice 1	Invoice identifier 1
Invoice 2	Invoice identifier 2
...	...

10 **[0079]** Assuming that only invoice 1 and invoice 2 are recorded in this blockchain, after an invalidation request that is initiated for invoice 1 and includes invoice identifier 1 is received, it can be determined, based on Table 2, that the blockchain stores invoice 1 corresponding to invoice identifier 1, and invoice 1 is determined as the target invoice; and after an invalidation request that is initiated for invoice 3 and
15 includes invoice identifier 3 is received, it can be determined, based on Table 2, that the blockchain does not store invoice 3 corresponding to invoice identifier 3; and so on.

[0080] In some illustrated implementations, because data stored in the blockchain cannot be tampered with, the target invoice can be invalidated by updating a state of
20 the target invoice from a valid state to an invalid state.

[0081] For example, when the target invoice is recorded in the blockchain, a state identifier for representing that the target invoice is valid can further be set for the target invoice, that is, the state of the target invoice is a valid state at this time. Subsequently, when it is determined that the target invoice needs to be invalidated, the
25 state identifier of the target invoice can be updated to represent that the target invoice is invalid, that is, the state of the target invoice is updated from a valid status to an invalid state.

[0082] The state identifier can be a specific byte set for the invoice, which is not limited in the present specification.

[0083] In the above technical solutions, when an invoice invalidation party initiates an invalidation request on a certain invoice recorded in a blockchain, corresponding logic in a smart contract deployed in a blockchain network of the blockchain is invoked to verify invalidation authority of the invoice invalidation party, and when it is determined that the invoice invalidation party has the authority to invalidate the invoice, the invoice is invalidated. As such, the invoice recorded in the blockchain is invalidated, and the data security of invoices circulating in the blockchain is ensured.

[0084] FIG. 2 is a flowchart illustrating another invoice invalidation method based on a blockchain according to some example implementations of the present specification. The method can be applied at a node device in a blockchain network, and includes the following steps.

[0085] Step 202: Obtain a transaction that has been recorded in the blockchain.

[0086] Step 204: Determine whether the transaction is a refund transaction corresponding to a payment transaction that has been recorded in the blockchain.

[0087] Step 206: In response to that the transaction is a refund transaction corresponding to the payment transaction, invoke an invoice invalidation logic provided in a smart contract deployed in a blockchain network of the blockchain to invalidate a target invoice that is related to the payment transaction and recorded in the blockchain.

[0088] In the implementations, after completing a payment or refund, a user can record the payment transaction or refund transaction in a blockchain.

[0089] A node device in the blockchain can obtain a transaction that has been recorded in the blockchain, and determine whether the transaction is a refund transaction corresponding to a certain payment transaction that has been recorded in the blockchain.

[0090] Specifically, newly-added transactions that have been recorded in the blockchain can be acquired regularly based on a certain time interval, and it is determined whether each transaction is a refund transaction corresponding to a certain payment transaction that has been recorded in the blockchain separately; alternatively or additionally, when a new transaction record is added in the blockchain, it can be immediately determined whether the transaction is a refund transaction corresponding to a certain payment transaction that has been recorded in the blockchain, which is not limited in the present specification.

[0091] In practice, it can be checked whether obtained transaction information such as transaction objects and transaction content matches corresponding transaction information of a payment transaction that has been recorded in the blockchain.

[0092] The transaction objects include a payer and a payee of a transaction, and transaction content can include transaction amount, transaction items, and the like.

[0093] If the payer of a transaction is the payee of another payment transaction that has been recorded in the blockchain, the payee of the transaction is the payer of the other payment transaction, and the transaction content of the transaction is the same as that of the payment transaction, it can be determined that the transaction is a refund transaction corresponding to the other payment transaction.

[0094] If it is determined that the obtained transaction is a refund transaction corresponding to a certain payment transaction that has been recorded in the blockchain, invoice invalidation logic provided in a smart contract deployed in the blockchain network can be invoked to invalidate a target invoice that is related to the payment transaction and recorded in the blockchain.

[0095] The invoice invalidation logic can be, for example, program code that is provided in the smart contract and related to execution logic for invalidating an invoice.

[0096] In practice, the above smart contract can further provide transaction matching logic. In this case, authority verification logic provided in the smart contract can be invoked to determine whether the above obtained transaction is a refund transaction corresponding to a certain payment transaction that has been recorded in the blockchain.

[0097] Transaction matching logic can specifically be program code that is provided in the smart contract and related to execution logic for matching with transaction information in a transaction.

[0098] It should be noted that the smart contract for invoice invalidation described above and the smart contract for matching with transaction information in a transaction can be integrated into one smart contract and deployed in the above blockchain, or can be deployed in the blockchain as two different smart contracts, which is not limited in the present specification.

[0099] In some illustrated implementations, after it is determined that the above obtained transaction is a refund transaction corresponding to a certain payment transaction that has been recorded in the above blockchain, based on an invoice

identifier corresponding to the transaction, it can be first determined whether a target invoice related to the payment transaction has been recorded in the blockchain, and when it is determined that the target invoice has been recorded in the blockchain, invoice invalidation logic provided in a smart contract deployed in the blockchain network, *e.g.*, stored in a blockchain of the blockchain network, can be invoked to invalidate the target invoice, so as to avoid invalid operation when the target invoice has not been recorded in the blockchain, and reduce device resource consumption.

[0100] For example, when an invoice issued for a certain payment transaction is recorded in the blockchain, the invoice identifier of the invoice can be correspondingly recorded in the blockchain. In addition, a corresponding relationship between the invoice identifier of the invoice and the payment transaction can further be recorded in the blockchain.

[0101] After it is determined that the refund transaction corresponding to the payment transaction is obtained, the invoice identifier corresponding to the payment transaction can be obtained, and it is checked whether the invoice identifier matches the invoice identifier of each invoice that has been recorded in the blockchain. In response to that the invoice identifier of the invoice is the same as an invoice identifier of a certain invoice that has been recorded in the blockchain, it can be determined that the blockchain stores an invoice corresponding to the invoice identifier, and the invoice is determined as the target invoice. Subsequently, the target invoice can be invalidated.

[0102] Accordingly, in response to that it is determined that the blockchain does not store an invoice corresponding to the invoice identifier, the target invoice that is related to the payment transaction and recorded in the blockchain no longer needs to be invalidated.

[0103] The invoice identifier can be a hash value obtained through hash calculation based on an invoice content; or a hash value obtained through hash calculation based on a piece of uniqueness information (such as an invoice number and an invoice code) in the invoice content.

[0104] In practice, the invoice issued for a certain payment transaction and the payment transaction can be directly recorded in the blockchain. As such, after it is determined that the refund transaction corresponding to the payment transaction is obtained, it can be directly determined whether the blockchain stores a target invoice related to the payment transaction.

[0105] In some illustrated implementations, because data stored in the blockchain cannot be tampered with, the target invoice can be invalidated by updating a state of the target invoice from a valid state or an invalid state.

[0106] For example, when the target invoice is recorded in the blockchain, a state identifier for representing that the target invoice is valid can further be set for the target invoice, that is, the state of the target invoice is a valid state at this time. Subsequently, when it is determined that the target invoice needs to be invalidated, the state identifier of the target invoice can be updated to represent that the target invoice is invalid, that is, the state of the target invoice is updated from a valid status to an invalid state.

[0107] The state identifier can be a specific byte set for the invoice, which is not limited in the present specification.

[0108] In the above technical solutions, when it is determined that a certain transaction that has been recorded in a blockchain is a refund transaction corresponding to another payment transaction that has been recorded in the blockchain, corresponding logic in a smart contract deployed in the blockchain network is invoked to invalidate an invoice that is related to the other payment transaction and recorded in the blockchain. As such, the invoice recorded in the blockchain is invalidated, and the data security of invoices circulating in the blockchain is ensured.

[0109] Corresponding to the above implementations of the invoice invalidation method based on a blockchain, the present specification further provides implementations of the invoice invalidation apparatus based on a blockchain.

[0110] The implementations of the invoice invalidation apparatus based on a blockchain according to the present specification can be applied to an electronic device. The apparatus implementations can be implemented by using software, hardware, or a combination thereof. Taking software implementation as an example, a logical apparatus is implemented by reading, by using a processor of an apparatus device where the apparatus is located, corresponding computer program instructions in a non-volatile memory into a memory for running. On the hardware level, FIG. 3 is a diagram illustrating a hardware structure of an electronic device where an invoice invalidation apparatus based on a blockchain is located according to the present specification. In addition to a processor, a memory, a network interface, and a non-volatile memory shown in FIG. 3, the electronic device where the apparatus

according to the implementations is located can generally further include other hardware depending on the actual invoice invalidation function based on a blockchain. Details are omitted for simplicity.

[0111] FIG. 4 is a block diagram illustrating an invoice invalidation apparatus based on a blockchain according to some example implementations of the present specification. The apparatus 40 can be applied to the electronic device shown in FIG. 3, and includes: a receiving module 401, configured to receive an invalidation request that is initiated by an invoice invalidation party for a target invoice recorded in the blockchain; a first determining module 402, configured to: in response to the invalidation request, determine whether the invoice invalidation party has authority to invalidate the target invoice; and an invalidation module 403, configured to: in response to that the invoice invalidation party has the authority to invalidate the target invoice, invoke an invoice invalidation logic provided in a smart contract deployed in a blockchain network of the blockchain to invalidate the target invoice.

[0112] In the implementations, the smart contract can further provide authority verification logic for the invoice invalidation party; and the first determining module 402 can be specifically configured to: invoke the authority verification logic provided in the smart contract to determine whether the invoice invalidation party has authority to invalidate the target invoice.

[0113] In the implementations, the invalidation request can include a user identifier of the invoice invalidation party; the first determining module 402 can be specifically configured to: determine, based on the user identifier, whether the invoice invalidation party is an issuer of the target invoice or a trusted third party; and in response to that the invoice invalidation party is an issuer of the target invoice or a trusted third party, determine that the invoice invalidation party has the authority to invalidate the target invoice.

[0114] In the implementations, the invalidation request can include an invoice identifier of the target invoice; the apparatus 40 can further include: a second determining module 404, configured to: determine, based on an invoice identifier, whether an invoice corresponding to the invoice identifier is stored in the blockchain; and a third determining module 405, configured to: when the invoice corresponding to the invoice identifier is stored in the blockchain, determine the invoice as the target invoice.

[0115] In the implementations, the invoice identifier can include: a hash value obtained through hash calculation based on an invoice content; or a hash value obtained through hash calculation based on a piece of uniqueness information in the invoice content.

5 [0116] In the implementations, the invalidation module 403 can be specifically configured to: update a state of the target invoice from a valid state to an invalid state.

[0117] FIG. 5 is a block diagram illustrating another invoice invalidation apparatus based on a blockchain according to some example implementations of the present specification. The apparatus 50 can be applied to the electronic device shown in FIG.
10 3, and includes: an acquisition module 501, configured to obtain a transaction that has been recorded in the blockchain; a first determining module 502, configured to determine whether the transaction is a refund transaction corresponding to a payment transaction that has been recorded in the blockchain; and an invalidation module 503, configured to: when the transaction is a refund transaction corresponding to the
15 payment transaction, invoke an invoice invalidation logic provided in a smart contract deployed in a blockchain network of the blockchain to invalidate a target invoice that is related to the payment transaction and recorded in the blockchain.

[0118] In the implementations, the smart contract can further provide transaction matching logic; and the first determining module 502 can be specifically configured
20 to: invoke the transaction matching logic provided in the smart contract to determine whether the transaction is a refund transaction corresponding to the payment transaction that has been recorded in the blockchain.

[0119] In the implementations, the apparatus 50 can further include: a second determining module 504, configured to: determine, based on an invoice identifier
25 corresponding to the payment transaction, whether an invoice corresponding to the invoice identifier is stored in the blockchain; and a third determining module 505, configured to: when the invoice corresponding to the invoice identifier is stored in the blockchain, determine the invoice as the target invoice.

[0120] In the implementations, the invoice identifier can include: a hash value
30 obtained through hash calculation based on an invoice content; or a hash value obtained through hash calculation based on a piece of uniqueness information in the invoice content.

[0121] In the implementations, the invalidation module 503 can be specifically configured to: update a state of the target invoice that is related to the payment transaction and recorded in the blockchain from a valid state to an invalid state.

[0122] The implementation process of functions of each module in the above apparatus is detailed in the implementation process of the corresponding steps in the above method. Details are omitted herein for simplicity.

[0123] Because the apparatus implementations basically correspond to the method implementations, for related parts, references can be made to partial description of the method implementation. The apparatus implementations described above are merely illustrative, where modules described as separate components can or cannot be physically separated. Components displayed as modules can or cannot be physical modules, that is, the components can be located in one place, or can be distributed to multiple network modules. Some or all of the modules can be selected based on actual needs to achieve the object of the solution of the present specification. A person of ordinary skill in the art can understand and implement the implementations without innovative efforts.

[0124] The systems, apparatuses, or modules illustrated in the above implementations can be specifically implemented by computer chips or entities, or by products with certain functions. A typical implementation device is a computer. The computer can be a personal computer, a laptop computer, a cellular phone, a camera phone, a smartphone, a personal digital assistant, a media player, a navigation device, an email sending and receiving device, a game console, a tablet computer, a wearable device, or a combination thereof.

[0125] Corresponding to the above implementations of the invoice invalidation method based on a blockchain, the present specification further provides implementations of an electronic device. The electronic device includes a processor and a memory configured to store machine-executable instructions, where the processor and the memory are usually connected to each other by using an internal bus. In other possible implementations, the device can further include an external interface that can be communicated with other devices or components.

[0126] In the implementations, by reading and executing the machine-executable instructions stored in the memory and corresponding to control logic for invoice invalidation based on a blockchain, the processor is enabled to: receive an invalidation request that is initiated by an invoice invalidation party for a target invoice recorded

in the blockchain; in response to the invalidation request, determine whether the invoice invalidation party has authority to invalidate the target invoice; and in response to that the invoice invalidation party has the authority to invalidate the target invoice, invoke an invoice invalidation logic provided in a smart contract deployed in
5 a blockchain network of the blockchain to invalidate the target invoice.

[0127] In the implementations, the smart contract further provides authority verification logic for the invoice invalidation party; and by reading and executing the machine-executable instructions stored in the memory and corresponding to control logic for invoice invalidation based on a blockchain, the processor is enabled to:
10 invoke the authority verification logic provided in the smart contract to determine whether the invoice invalidation party has authority to invalidate the target invoice.

[0128] In the implementations, the invalidation request includes a user identifier of the invoice invalidation party; by reading and executing the machine-executable instructions stored in the memory and corresponding to control logic for invoice
15 invalidation based on a blockchain, the processor is enabled to: determine, based on the user identifier, whether the invoice invalidation party is an issuer of the target invoice or a trusted third party; and in response to that the invoice invalidation party is an issuer of the target invoice or a trusted third party, determine that the invoice invalidation party has the authority to invalidate the target invoice.

[0129] In the implementations, the invalidation request includes an invoice identifier of the target invoice; by reading and executing the machine-executable instructions stored in the memory and corresponding to control logic for invoice
20 invalidation based on a blockchain, the processor is further enabled to: determine, based on an invoice identifier, whether an invoice corresponding to the invoice identifier is stored in the blockchain; and in response to that the invoice corresponding to the invoice identifier is stored in the blockchain, determine the invoice as the target
25 invoice.

[0130] In the implementations, the invoice identifier can include: a hash value obtained through hash calculation based on an invoice content; or a hash value
30 obtained through hash calculation based on a piece of uniqueness information in the invoice content.

[0131] In the implementations, by reading and executing the machine-executable instructions stored in the memory and corresponding to control logic for invoice

invalidation based on a blockchain, the processor is enabled to: update a state of the target invoice from a valid state to an invalid state.

[0132] Corresponding to the above implementations of the invoice invalidation method based on a blockchain, the present specification further provides an implementation of another electronic device. The electronic device includes a processor and a memory configured to store machine-executable instructions, where the processor and the memory are usually connected to each other by using an internal bus. In other possible implementations, the device can further include an external interface that can be communicated with other devices or components.

[0133] In the implementations, by reading and executing the machine-executable instructions stored in the memory and corresponding to control logic for invoice invalidation based on a blockchain, the processor is enabled to: obtain a transaction that has been recorded in the blockchain; determine whether the transaction is a refund transaction corresponding to a payment transaction that has been recorded in the blockchain; and in response to that the transaction is a refund transaction corresponding to the payment transaction, invoke an invoice invalidation logic provided in a smart contract deployed in a blockchain network of the blockchain to invalidate a target invoice that is related to the payment transaction and recorded in the blockchain.

[0134] In the implementations, the smart contract further provides transaction matching logic; and by reading and executing the machine-executable instructions stored in the memory and corresponding to control logic for invoice invalidation based on a blockchain, the processor is enabled to: invoke the transaction matching logic provided in the smart contract to determine whether the transaction is a refund transaction corresponding to the payment transaction that has been recorded in the blockchain.

[0135] In the implementations, by reading and executing the machine-executable instructions stored in the memory and corresponding to control logic for invoice invalidation based on a blockchain, the processor is further enabled to: determine, based on an invoice identifier corresponding to the payment transaction, whether an invoice corresponding to the invoice identifier is stored in the blockchain; and in response to that the invoice corresponding to the invoice identifier is stored in the blockchain, determine the invoice as the target invoice.

[0136] In the implementations, the invoice identifier can include: a hash value obtained through hash calculation based on an invoice content; or a hash value obtained through hash calculation based on a piece of uniqueness information in the invoice content.

5 **[0137]** In the implementations, by reading and executing the machine-executable instructions stored in the memory and corresponding to control logic for invoice invalidation based on a blockchain, the processor is enabled to: update a state of the target invoice that is related to the payment transaction and recorded in the blockchain from a valid state to an invalid state.

10 **[0138]** A person skilled in the art will easily figure out other implementation solutions of the present specification after considering the present specification and practicing the invention disclosed herein. The present specification is intended to cover any modification, use or adaptive change of the present specification, and these modifications, uses or adaptive changes follow the general principles of the present
15 specification and include common knowledge or conventional technical means in the technical field not disclosed in the present specification. The present specification and implementations are regarded as example only, with the true scope and spirit of the present specification being indicated by the following claims.

[0139] It should be understood that the present specification is not limited to the
20 precise structure already described above and shown in the accompanying drawings, and various modifications and changes can be performed without departing from its scope. The scope of the present specification is limited only by the appended claims.

[0140] The above descriptions are merely the preferred implementations of the present specification, and are not intended to limit the present specification. Any
25 modifications, equivalent replacements, improvements, and the like made within the spirit and principles of the present specification should fall within the protection scope of the present specification.

CLAIMS

What is claimed is:

1. An invoice invalidation method based on a blockchain, comprising:
receiving an invalidation request that is initiated by an invoice invalidation
5 party for a target invoice recorded in the blockchain;
in response to the invalidation request, determining whether the invoice
invalidation party has authority to invalidate the target invoice; and
in response to that the invoice invalidation party has the authority to invalidate
the target invoice, invoking an invoice invalidation logic provided in a smart contract
10 deployed in a blockchain network of the blockchain to invalidate the target invoice.

2. The method according to claim 1, wherein:
the smart contract further provides an authority verification logic for the
invoice invalidation party; and
15 the determining whether the invoice invalidation party has authority to
invalidate the target invoice includes:
invoking the authority verification logic provided in the smart contract
to determine whether the invoice invalidation party has authority to invalidate the
target invoice.

20 3. The method according to claim 1, wherein:
the invalidation request includes a user identifier of the invoice invalidation
party; and
the determining whether the invoice invalidation party has authority to
25 invalidate the target invoice includes:
determining, based on the user identifier, whether the invoice
invalidation party is an issuer of the target invoice or a trusted third party; and
in response to that the invoice invalidation party is an issuer of the
target invoice or a trusted third party, determining that the invoice invalidation party
30 has the authority to invalidate the target invoice.

4. The method according to claim 1, wherein:
the invalidation request includes an invoice identifier of the target invoice; and

the method further includes:

determining, based on an invoice identifier, whether an invoice corresponding to the invoice identifier is stored in the blockchain; and

in response to that the invoice corresponding to the invoice identifier is
5 stored in the blockchain, determining the invoice as the target invoice.

5. The method according to claim 4, wherein the invoice identifier includes:

a hash value obtained through hash calculation based on an invoice content; or

a hash value obtained through hash calculation based on a piece of uniqueness
10 information in the invoice content.

6. The method according to claim 1, wherein the invalidating the target invoice includes:

updating a state of the target invoice from a valid state to an invalid state.

15

7. An invoice invalidation method based on a blockchain, comprising:

obtaining a transaction that has been recorded in the blockchain;

determining whether the transaction is a refund transaction corresponding to a payment transaction that has been recorded in the blockchain; and

20 in response to the transaction is a refund transaction corresponding to the payment transaction, invoking an invoice invalidation logic provided in a smart contract deployed in a blockchain network of the blockchain to invalidate a target invoice that is related to the payment transaction and recorded in the blockchain.

25 8. The method according to claim 7, wherein:

the smart contract further provides a transaction matching logic; and

the determining whether the transaction is a refund transaction corresponding to the payment transaction that has been recorded in the blockchain includes:

invoking the transaction matching logic provided in the smart contract
30 to determine whether the transaction is a refund transaction corresponding to the payment transaction that has been recorded in the blockchain.

9. The method according to claim 7, further including:

determining, based on an invoice identifier corresponding to the payment transaction, whether an invoice corresponding to the invoice identifier is stored in the blockchain; and

in response to that the invoice corresponding to the invoice identifier is stored
5 in the blockchain, determining the invoice as the target invoice.

10. The method according to claim 9, wherein the invoice identifier includes:
a hash value obtained through hash calculation based on an invoice content; or
a hash value obtained through hash calculation based on a piece of uniqueness
10 information in the invoice content.

11. The method according to claim 7, wherein the invalidating the target invoice that is related to the payment transaction and recorded in the blockchain includes:
15 updating a state of the target invoice that is related to the payment transaction and recorded in the blockchain from a valid state to an invalid state.

12. An invoice invalidation apparatus based on a blockchain, comprising:
a receiving module, configured to receive an invalidation request that is
20 initiated by an invoice invalidation party for a target invoice recorded in the blockchain;
a first determining module, configured to: in response to the invalidation request, determine whether the invoice invalidation party has authority to invalidate the target invoice; and
25 an invalidation module, configured to: in response to that the invoice invalidation party has the authority to invalidate the target invoice, invoke an invoice invalidation logic provided in a smart contract deployed in a blockchain network of the blockchain to invalidate the target invoice.

30 13. The apparatus according to claim 12, wherein:
the smart contract further provides authority verification logic for the invoice invalidation party; and
the first determining module is specifically configured to:

invoke the authority verification logic provided in the smart contract to determine whether the invoice invalidation party has authority to invalidate the target invoice.

5 14. The apparatus according to claim 12, wherein:
the invalidation request includes a user identifier of the invoice invalidation party; and
the first determining module is specifically configured to:
determine, based on the user identifier, whether the invoice
10 invalidation party is an issuer of the target invoice or a trusted third party; and
in response to the invoice invalidation party is an issuer of the target invoice or a trusted third party, determine that the invoice invalidation party has the authority to invalidate the target invoice.

15 15. The apparatus according to claim 12, wherein:
the invalidation request includes an invoice identifier of the target invoice; and
the apparatus further includes:
a second determining module, configured to: determine, based on an
invoice identifier, whether an invoice corresponding to the invoice identifier is stored
20 in the blockchain; and
a third determining module, configured to: in response to that the invoice corresponding to the invoice identifier is stored in the blockchain, determine the invoice as the target invoice.

25 16. The apparatus according to claim 15, wherein the invoice identifier includes:
a hash value obtained through hash calculation based on invoice content; or
a hash value obtained through hash calculation based on a piece of uniqueness
information in the invoice content.

30 17. The apparatus according to claim 12, wherein the invalidation module is specifically configured to:
update a state of the target invoice from a valid state to an invalid state.

18. An invoice invalidation apparatus based on a blockchain, comprising:
an acquisition module, configured to obtain a transaction that has been recorded in the blockchain;

5 a first determining module, configured to determine whether the transaction is a refund transaction corresponding to a payment transaction that has been recorded in the blockchain; and

an invalidation module, configured to: in response to that the transaction is a refund transaction corresponding to the payment transaction, invoke an invoice invalidation logic provided in a smart contract deployed in a blockchain network of
10 the blockchain to invalidate a target invoice that is related to the payment transaction and recorded in the blockchain.

19. The apparatus according to claim 18, wherein:

the smart contract further provides transaction matching logic; and

15 the first determining module is specifically configured to:

invoke the transaction matching logic provided in the smart contract to determine whether the transaction is a refund transaction corresponding to the payment transaction that has been recorded in the blockchain.

20 20. The apparatus according to claim 18, further including:

a second determining module, configured to: determine, based on an invoice identifier corresponding to the payment transaction, whether an invoice corresponding to the invoice identifier is stored in the blockchain; and

25 a third determining module, configured to: when the invoice corresponding to the invoice identifier is stored in the blockchain, determine the invoice as the target invoice.

21. The apparatus according to claim 20, wherein the invoice identifier includes:

30 a hash value obtained through hash calculation based on invoice content; or

a hash value obtained through hash calculation based on a piece of uniqueness information in the invoice content.

22. The apparatus according to claim 18, wherein the invalidation module is specifically configured to:

update a state of the target invoice that is related to the payment transaction and recorded in the blockchain from a valid state to an invalid state.

5

23. An electronic device, comprising:

a processor; and

a memory configured to store machine-executable instructions,

wherein by reading and executing the machine-executable instructions stored
10 in the memory and corresponding to control logic for invoice invalidation based on a blockchain, the processor is enabled to:

receive an invalidation request that is initiated by an invoice
invalidation party for a target invoice recorded in the blockchain;

in response to the invalidation request, determine whether the invoice
15 invalidation party has authority to invalidate the target invoice; and

in response to that the invoice invalidation party has the authority to
invalidate the target invoice, invoke an invoice invalidation logic provided in a smart
contract deployed in a blockchain network of the blockchain to invalidate the target
invoice.

20

24. An electronic device, comprising:

a processor; and

a memory configured to store machine-executable instructions,

wherein by reading and executing the machine-executable instructions stored
25 in the memory and corresponding to control logic for invoice invalidation based on a blockchain, the processor is enabled to:

obtain a transaction that has been recorded in the blockchain;

determine whether the transaction is a refund transaction
corresponding to a payment transaction that has been recorded in the blockchain; and

in response to the transaction is a refund transaction corresponding to
30 the payment transaction, invoke an invoice invalidation logic provided in a smart
contract deployed in a blockchain network of the blockchain to invalidate a target
invoice that is related to the payment transaction and recorded in the blockchain.