



Designing Smart Contract for Electronic Document Taxation

Dimaz Ankaa Wijaya^{1,2(✉)}, Joseph K. Liu¹, Ron Steinfeld¹, Dongxi Liu²,
Fengkie Junis³, and Dony Ariadi Suwarsono^{3,4}

¹ Monash University, Melbourne, Australia

{dimaz.wijaya, joseph.liu, ron.steinfeld}@monash.edu

² Data61, CSIRO, Canberra, Australia

dongxi.liu@data61.csiro.au

³ Gadjah Mada University, Yogyakarta, Indonesia

fengkie.junis@mail.ugm.ac.id

⁴ Directorate General of Taxes, Jakarta, Indonesia

dony.suwarsono@pajak.go.id

Abstract. In recent years, we have seen a massive blockchain adoption in cryptocurrencies such as Bitcoin. Following the success of blockchain in cryptocurrency industry, many people start to explore the possibility of implementing blockchain technology in different fields. In this paper, we propose Smart Stamp Duty, a system which can revolutionize the way stamp duty (document tax) is managed and paid. The proposed Smart Stamp Duty offers significant improvements on the convenience when paying stamp duty by removing the need of physical revenue stamp. At the same time, the blockchain technology also provides the auditability of the recorded data. Smart stamp duty enables the expansion of the existing electronic stamp duty application to retail level. Smart stamp duty also enables the taxpayers to pay the stamp duty of their electronic documents. Our proposed system also enables the taxpayers to convert their electronic documents into physical documents while still maintain the ability to refer the electronic-based stamp duty payments.

Keywords: Blockchain · Smart contract · Electronic document · Taxation

1 Introduction

According to 2018 Indonesia State Budget, taxation is the major source of state revenue, comprising 1.6 quadrillion Rupiah. The number has sharply increased from 1.47 quadrillion Rupiah in 2017 [9–11, 15]. The increase of taxation target in 2018 is due to several assumptions, including automatic exchange of information (AEOI), tax incentives, and improvements on human resources, information technology, and taxation services to the taxpayers [10].

Stamp duty is a state tax levied on specific types of documents. Stamp duty is categorised as “Other Taxes” category in Indonesia’s with a total revenue target

of 9 trillion Rupiahs [9]. This target is about 0.6% of the overall 2018 revenue, increasing 11% from the 2017 state budget. Despite the small percentage the stamp duty target carries, Directorate General of Taxes (DGT) as the Indonesian Tax Authority looks for revenue improvements in this area [17]. Online transactions should be evaluated as the users might also need to pay stamp duties on the electronic documents, aside from the paper-based documents, involved in the business [21].

1.1 Problem Definition

Indonesian stamp duty is currently only levied on paper-based documents, as defined in the Stamp Duty Act 1985 [13]. The taxpayers need to attach physical revenue stamps, sign them properly, write the dates, and only then the tax payments are deemed valid. The problem with physical stamp is that it is inconvenient to get, because the taxpayers need to buy the physical stamps on brick-and-mortar stores to be attached on their physical documents.

There is also a limited use of computerized stamp duty which is only valid under certain circumstances [16]. To use the computerized stamp duty, the taxpayers need to purchase tax credits. The amount of the tax credits to be purchased is based on their estimated number of documents they will create in the following month. However, not everyone is allowed to use the computerized stamp duty. Only those who create at least one hundred documents per day subject to stamp duty are allowed to use the mechanism.

The existing stamp duty mainly focuses on physical documents; hence, the physical stamp is required. However, as technology advances, computerization changes the way people conduct their business. Electronic documents are more convenient to create, manage, and process compared to physical documents. Although these electronic documents may be owed stamp duty, it is infeasible for the taxpayers to purchase physical stamps and attach them to the electronic documents [5].

In summary, we identify the gap of using computerized stamp duty where individuals or companies that prefer to use electronic documents where these documents are taxable by the stamp duty regulations. Several solutions have been proposed to solve the problem [7, 21]. However, the usability and correctness of the data stored in the proposed solutions are questionable, where the payment proofs are stored in traditional information systems without any proper auditable features by all parties.

1.2 Our Contribution

Considering the current limitation of both physical and electronic stamp duty mechanism, we propose a new blockchain-based electronic stamp duty as an effort to increase the stamp duty revenue by using technological advancement. The system will increase the convenience of the taxpayers to pay the stamp duty when creating taxable documents. The system can be utilised by all taxpayers through information technology devices. The system will also provide a public

verifiability to prove that an electronic document has the stamp duty paid. The built-in cryptographic techniques also prevent forgery which happened in the physical stamp duty [8].

We coin the term **Smart Stamp Duty** to refer to our proposed system which utilises smart contract as well as blockchain technology to record information and conduct predetermined business processes automatically based on the user's inputs and program logic. The smart contract is fully auditable, transparent to its stakeholders, and the finality of the data is guaranteed. Moreover, its standard features such as timestamp and digital signature fulfil the requirements of a valid stamp duty. Smart contract is scalable; not only retail users but also wholesale users can use the **Smart Stamp Duty** as long as there exist sufficient applications to pose as user interfaces to communicate with the smart contract.

2 Background

2.1 Indonesian Stamp Duty

Indonesian Stamp Duty Act 1985 determines that there are documents (in the physical form) owed stamp duty [13]. As a proof that the tax is paid, a form of physical stamp is glued on the documents and signed [13]. The cost of the stamp depends on the category of the document and the amount of money printed on the document [13].

2.2 Blockchain

Blockchain was first applied in Bitcoin, a decentralized payment system introduced by Satoshi Nakamoto [12]. The blockchain technology enables users to create transactions without the need of proving their real identities. There is no central authority controlling the system. Instead every user can verify and validate transactions data by using a set of protocols.

Blockchain can be divided into three models based on its type of participants, namely public blockchain, private blockchain, and consortium blockchain [18]. Public blockchain is a type of blockchain which allows everyone to participate at any time and allows anyone to leave whenever they desire. A private blockchain, as its name implies, is a closed system where the participants need specific permissions from a central authority which controls the whole system. Although blockchain is originally intended to remove the central authority, however in some cases the central authority is still required and cannot be removed, for example due to legal requirements.

The last blockchain model is consortium blockchain. It is a combination between public and private blockchain, where there will be several authorities sharing the power equally among them to control the system, however the membership of the authorities is not open as in a public blockchain. Consortium blockchain also applies access control mechanism where not everyone has access to the blockchain.

To replace the role of a central authority, blockchain utilises consensus mechanism to determine the canonical version of the record and rules who may write new information for a given block. Proof-of-Work (PoW) and proof-of-stake (PoS) are two most common consensus mechanisms for public blockchains. Consortium or private blockchain might use different methods of consensus algorithm that meet the system requirement.

2.3 Smart Contract

While Bitcoin's blockchain has a limited set of operation codes (*opcodes*), smart contract platforms such as Ethereum was created to support Turing-complete application in the blockchain [20]. Smart contract is generally an application stored in the blockchain. It receives inputs from the user, runs functions, and generates outputs. The execution of the smart contract codes are generally executed on-chain by the nodes of the network. As it is stored in the blockchain, the codes are permanent and transparent, hence it offers fairness to the participants. A smart contract platform also inherits the same traits as other blockchain systems, where the transaction data is visible and verifiable to anyone.

3 Related Work

3.1 Electronic Stamp Duty

The business model of electronic stamp duty is investigated by [7, 21]. The Information and Electronic Transaction Act 2008 has provided a foundation in which the term document can be extended, not only in the physical form but also in electronic form, since both forms are now considered as legal proofs in Indonesian jurisdiction [14]. Before the act was established, all legal proofs must be printed and legalized, but now all electronic information no longer need to be printed in physical form, and if the information is properly handled, it can be a legal proof.

3.2 Blockchain Technology in Taxation

Blockchain does not only show its massive potential in the payment industry, but also other areas such as taxation. Previous studies describe that blockchain technology can mitigate the problem of tax losses due to international trades [1–4]. The blockchain technology has been proposed to modernize Value-Added Tax (VAT) [19]. In this case, the blockchain is utilized to enhance the information openness across multiple authorities and, likewise, to share information between those authorities.

4 Smart Stamp Duty

4.1 Overview

Our system works in the blockchain environment where cryptographic techniques such as digital signature and hash functions exist. We will define the term tokens to refer to electronic money balance owned by a user. The tokens can be used to pay the stamp duty in a smart contract provided by the Tax Authority. Before a user pays the stamp duty, she needs to buy the tokens before paying the stamp duty for each taxable document she creates. The transparency of the system enables the actors of the system to verify and validate all information stored in the blockchain.

4.2 The Blockchain

The blockchain as a shared database which can be used to store the data as well as the smart contracts, and then run the smart contracts based on the input provided by the users. In the proposed system, we use the consortium-based permissioned blockchain model where the Tax Authority along with other trusted participants such as government's auditing bodies can validate new transactions and secure the blockchain.

There will be multiple nodes running the system. These nodes can be managed by multiple participants, for example banks, the tax authority, and the monitoring bodies. Each of these participants keeps a copy of all transactions in the system for auditability and accountability purposes. Although there can be many nodes owned by different participants, the access rights will still need to be set up by the Tax Authority as the system administrator.

4.3 The Smart Contract

The smart contract is a set of permanent codes stored in the blockchain. The smart contract also stores information and runs application logic based on the previously stored codes. The purpose of the smart contract is to make the application logic transparent, auditable, and to let everyone sees information stored in the smart contract. The smart contract of our solution will have several parameters to be set by the owner based on the regulations. The smart contract of our solution is created and owned by the Tax Authority.

4.4 The Roles

We determine several roles involved in the system: the users, the bank, the Tax Authority, and monitoring organisations. Each role has its own credentials to do different activities.

The Taxpayers. The taxpayers are using the system to pay their tax dues, e.g. stamp duty. They interact with the system by using application interfaces provided. The taxpayers interact with the banks or any retail sellers to buy balance by using electronic money, credit cards, bank accounts, or cash. The users produce the documents, pay the stamp duty, and send the documents to other users. The users can also verify that the documents they receive have the tax paid.

The Bank. The bank usually provides financial services to the users. In the proposed system, the bank also provides a gateway to convert the local currency to tokens to access the system. The bank ensures that the exchange system is accessible by the taxpayers whenever they want to buy the tokens to be used to pay the stamp duty. At the end of a predetermined period, the bank reports to the tax authority regarding the amount of money collected from the taxpayers and send all the money to the government's account.

The Tax Authority. The Tax Authority has the authority to determine the tax rate to be paid by the taxpayers. In our system, the Tax Authority controls the blockchain system, including deploying or modifying the smart contracts used to run the system. The Tax Authority could authorize other parties to get involved in the system.

Tax Monitoring Organisations. The tax monitoring organisations help running the system by participating the closed consensus mechanism. In this mechanism, these organisations along with the Tax Authority determine the transactions to be stored in the blockchain. These organisations can have access to the system and the information stored in it.

4.5 The Applications

The applications are the interface between the users and the blockchain/smart contract. The applications used by different actors can be different based on their own authority. The application used by The Tax Authority may have the ability to modify the variables (by using the appropriate private keys), but the users can only add new records.

There can also be some additional applications to conduct the business, such as word extractor which enables the user to create the appropriate Bloom Filter to supply the stamp duty payment. Bloom Filter checker will also be required for validation in case the user prints the electronic documents and the printed documents require verification. However, these additional applications are not discussed in this paper.

4.6 The Token

Tokens are numbers stored in a smart contract and can be moved from an address to other addresses. The tokens are not intended to replace Indonesian national fiat currency as it will violate Indonesian Currency Act 2011. Tokens are used as proofs that the taxpayers have purchased tax credits which can be used to pay stamp duty. The tax credits are in the form of the balance of addresses. Only one token type will be created and available in the **Smart Stamp Duty** system, where the number of tokens can be increased or reduced by the Tax Authority.

4.7 Core Business Process

The core business process is shown in Fig. 1. A simple description for each activity is as follows.

1. User A creates an electronic document (e-document) subject to stamp duty. The electronic document can be in any form (word processor, spreadsheet, raw data, etc).
2. User A purchases tokens from a bank which can be used to pay the stamp duty. The user informs the bank of her “address”.
3. The bank credits the tokens to the user’s address by sending the correct transaction to the smart contract. The token balance can be checked by User A.
4. User A converts the electronic document into a protected format (i.e. PDF format), then computes the hash value of the e-document. The user then creates a transaction to put the hash value of the document to the smart contract. The transaction will decrease User A’s token balance based on the stamp duty paid.
The original document that has been signed needs to be kept by User A, while a new copy of the original document can be modified to include the receipt of the transaction.
5. The bank routinely provides a report to the Tax Authority regarding the total amount of balance purchased by taxpayers. The report can also be sent automatically or by using a specific interface.
6. The Tax Authority rechecks the report received from the bank to the actual transactions in the smart contract. The process can also be converted into an automatic job.
7. User A sends both the original copy and the paid copy of the electronic document to User B.
8. User B checks whether the stamp duty has been paid by querying the blockchain.

New users can join the system immediately by creating new public key pairs. All transactions can be read by all actors, thus reduce the risk of fraud.

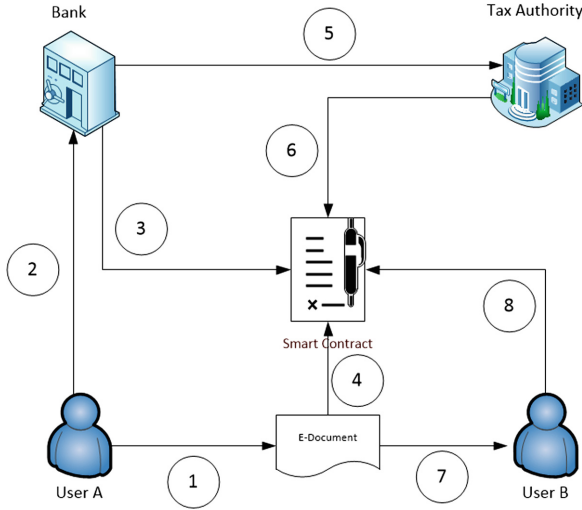


Fig. 1. The core business process of the Smart Stamp Duty.

4.8 The Parameters

Based on the business process we have defined, we then determine the detailed information to be stored in the smart contract. There are 3 types of the data: account parameters, system parameters, and transaction parameters.

Account Parameters. Account parameters are information regarding the token balance of the users. A new account will be created for a new user containing at least the user’s public key and an initial token balance (can be zero). The public key cannot be modified while the token balance will always be recalculated by the smart contract based on new transaction parameters.

System Parameters. System parameters are information required to run the system. The smart contract looks up to the system parameters before executing transactions. The smart parameters are determined by the Tax Authority based on the existing regulations. The parameters can always be changed whenever the regulation changes. The system parameters contain at least the following:

- The cost of the stamp.
- The regulation references.

The system parameters can be stored on the same script of the main smart contract or be put on a different script.

Transaction Parameters. Transaction parameters are information regarding stamp duty transaction created by the users. To satisfy the existing regulation

on stamp duty, the information items required for a valid transaction are the following:

- The public key of the taxpayer.
- The amount of the tax paid.
- Timestamp of the document.
- Hash value of the document.
- The payer’s signature.

In the real world, it might be possible for the taxpayers to create identical documents multiple times, hence they produce an identical hash value. Thus, to minimize the occurrence, it is possible to add unique information into the documents, such as public keys of the taxpayers or their tax file number and the document creation time.

4.9 The Program Logic

Based on the business process and the parameters, it is possible to determine the program logic needed to run the system. All the program logic are run by the smart contract independently.

System Parameter Definition. If the regulation changes, then the Tax Authority can change the system parameter by sending a new transaction. The smart contract checks if all required data is satisfied by the new transaction then stores the new parameter. The system parameters are defined in Sect. 4.8.

Token Transaction. Activities related to tokens are: token creation, token distribution and token sales, and lastly stamp duty payment.

- *Token creation.* As a central controller, Tax Authority is the only party that is given the authority to create tokens. The number of tokens to be created depends on the need. It is also possible to “mint” or create more tokens as required. However, during contract creation, token amount can also be initialized.
- *Token distribution and token sales.* Based on existing regulations, the Tax Authority can distribute the tokens to authorized participants such as banks and post office. The authorized participants can sell the tokens to any secondary markets such as distributors and retailers or sell them directly to end users (taxpayers).
- *Stamp duty payment.* If the user wants to pay the stamp duty, then the user creates a new transaction to do so. The smart contract first checks if there is enough balance in the user’s account, checks if required data is supplied. If everything is satisfied, then the smart contract reduces the amount of the stamp duty to be paid in the user’s balance before storing the document data in the smart contract. To prove that the user has paid the cost, the tokens will be sent to the Tax Authority’s address. The tokens will be accumulated by the Tax Authority which will be used for audit or reconciliation when calculating the real number of tax paid in fiat currency.

Tax Payment Audit. For the purpose of audit, the tokens received by the Tax Authority will not be recirculated. When the audit is finished, then the same number of tokens verified to be the stamp duty revenue could be sent to an unusable address (an address without known private key) such as zero address.

The scope of the audit can be expanded such that auditing bodies can directly assess and monitor the system and evaluate the data in a real time. By using blockchain as a shared ledger, the amount of token that has been sold and the amount of token that has been spent will be easy to calculate.

4.10 Transaction Fee

Assuming the blockchain will run in a permissioned environment, then it is possible to waive transaction fees for all transactions created within the system. It is assumed that the Tax Authority will not abuse the system by flooding the system by sending system parameter-defining transactions. It is also assumed that when creating transactions, the Bank and the User need to have enough balance to pay the stamp duty. Without a proper information supply, the transaction will not be processed and confirmed by the system.

However, without any transaction fee, the system is prone to abuse by participants, for example creating many transactions to their own addresses to slow down the entire system. To avoid the useless transaction flooding, a small transaction fee can be introduced such that creating many transactions will be costly to the attacker. The implementation of transaction fee is beyond the scope of this paper.

4.11 Security Model

We define two possible attacks to the proposed system as follows.

1. An adversary tries to mint new token without authorization.
2. Malicious bank tries to reduce the report provided to Tax Authority to keep the tax money illegally.

In a system where Tax Authority is assumed to be always honest and no security vulnerability is found on the blockchain system and assuming that the cryptographic techniques are secure, then:

1. The probability of an adversary to mint their own token is negligible; a token can only be minted by authorized party, which is Tax Authority.
2. The probability of a malicious bank modifying the report without being detected is also negligible, given the auditability feature of the blockchain.

5 Implementation and Evaluation

We have implemented our solution which can be deployed on Ethereum blockchain. Ethereum-like private blockchain environment such as Quorum¹ is preferred during implementation phase because of the following consideration.

¹ <https://www.jpmorgan.com/country/US/EN/Quorum>.

- A wide number of contributors and software developers that develop and use Ethereum system provide confidence for future improvements.
- Much peer-reviewed research has been conducted on Ethereum in terms of its security. Its security vulnerabilities have been well-discussed and documented. This provides a great assistance in developing secure system.
- Ethereum system has endured severe attacks and exploits since the first launch and proves its durability.

Our solution consists of two parts: the smart contract and the user interface. The smart contract code is available on Github². The user interface was written in Python and it is also available on Github³.

The smart contract uses a standard ERC-20 as its foundation which adds the capability of being traded on ERC-20 token markets due to its compatibility. We call the token **Smart Stamp Duty** (SSD). As with other ERC-20 tokens, our SSD is transferrable. Before paying the stamp duty, the user needs to buy the SSD token. Aside from SSD token, the native token Ether also exists in Ethereum environment. The transaction fee is paid by using Ether, while the amount of the fee is determined by calculating the complexity of the smart contract, expressed by gas. The gas is then converted into Ether by determining a conversion rate called gas cost or gas price. For simplification, the gas cost can be set to 0 Ether such that paying Ether will not be necessary in the implementation.

The main component, class **StampDuty**, contains two important data structures to contain information related to our solution, namely **StampParam** and **PayParam**. Each structure will be further explained.

5.1 Implementing Data Structures

StampParam. **StampParam**, contains information about types of stamp duty. Currently, there are only two stamps available which have different denominations: “*Rp3000*” and “*Rp6000*”⁴. Below is the detail of the **StampParam** structure.

- **StampCode**, to store stamp primary key (unique).
- **StampName**, to store human-readable stamp information.
- **StampPrice**, to store the price of the stamp.
- **RegulationReference**, to store the regulation reference of the stamp.
- **IsActive**, to flag active and inactive stamps.

StampParam is the master table for the stamp duty. Its importance is shown by only allowing the owner (i.e. the Tax Authority) to add and modify the data.

² <https://github.com/sonicskye/smart-stamp-duty>.

³ <https://github.com/sonicskye/smart-stamp-duty-ui>.

⁴ *Rp* stands for Rupiah, the Indonesian national currency.

PayParam. The second structure, `PayParam`, contains stamp duty payment data. A user adds a `PayParam` information every stamp duty paid. The `PayParam` structure is shown below.

- `PayCode`, works as the primary key of the payment (unique).
- `DocHash`, stores the hash value of the document related to the paid stamp.
- `PayIndex`, to store the index of the payment.
- `Payer`, to store the payer address.
- `StampCode`, to refer to the stamp paid by the transaction.
- `BloomFilter`, to store Bloom Filter information for testing.
- `TimeStamp`, to store timestamp in integer format.
- `TimeStampStr`, to store timestamp information in string format.
- `PayerSignature`, to store the payer’s signature.

As a unique key, the `PayCode` can be generated by a hash function H of multiple information, such as `DocHash`, `Payer`, `StampCode`, and `TimeStamp` when the document is submitted such that:

$$PayCode = H(DocHash||Payer||StampCode||TimeStamp)$$

where the symbol (`||`) is concatenation operation. `PayerSignature` is generated by a signing function $SIGN$ using `Ethereum`’s library under the following formula:

$$PayerSignature = SIGN(DocHash, PrivateKey)$$

where `PrivateKey` is the private key owned by the payer, associated to the payer’s address which is stored in `Payer` variable. Aside from the two data structures, there are also built-in functions to manage the system, including token minting and token burning to control the token supply.

Bloom Filter [6] value of the document is also recorded on the smart contract. It supports content matching mechanism when printing the electronic document into paper-based document. The value `BloomFilter` can be used to verify that the electronic and paper-based versions are identical. When computing the `BloomFilter`, the system will list all words in the document under a parameter set, including number of words. The verification is done by sampling keywords from the document. If the verification passes, then we can conclude that the paper-based document is identical to the electronic version of it.

5.2 Implementing Content Matching

The content matching feature is implemented in our source code. The feature simply tests a set of input strings which comes from a document against a Bloom Filter value taken from the claimed electronic stamp duty. The input can either be done manually or automatically, where the topic is beyond the scope of the paper. The content matching protocol is as follows.

- Determine the document W and its associated Bloom Filter value `BloomFilter` which can be retrieved by querying the smart contract using `PayCode`.
- Extract all distinct words w from the input document W such that $W = \{w_1, w_2, w_3, \dots, w_n\}$ where n is the number of distinct words.
- For each word w , check whether $w \in \text{BloomFilter}$. Store the result in R .
- The result is the percentage of *True* in R against the value n .
- The expected value is 100%. This result determines that there is a high probability that the document W is identical to the original document. Due to the false positive rate in Bloom Filter, it can never be confirmed that the examined document is identical to the original document.

6 Evaluation

6.1 System Evaluation

We deployed our implementation by using `Truffle` and `Ganache` as development environment in a private Ethereum network. The `PayParam` contains data duplicates such as timestamp and the payer's signature which are actually embedded in the transaction data and in the block where the transactions are included. However, reading the timestamp and signature directly from the block requires enormous computing power. When we tried this, the `Ganache` which we used as a node in our development environment crashed every time the code was tested.

To reduce the computing resources to read the transaction signatures and the block's timestamp, `PayParam` explicitly includes the payer's signature and data creation timestamp such that they can be read directly from the smart contract. As the result of this extra information, the required transaction fee doubled from around 235,000 `gas` to more than 450,000 `gas`.

6.2 Security Evaluation

Our system is designed to be implemented in Ethereum-like private blockchain environment where the Tax Authority poses as the central authority to control the whole system. It is assumed that the Tax Authority behaves honestly and never gives the tokens for free to any users and never leaks the private keys to the adversary. It is also assumed that the system does not have severe bugs which can be exploited by the adversary. In this scenario, it is infeasible for the adversary to create `SSD` tokens without a proper authorisation from the Tax Authority.

7 Conclusion and Future Work

We have described a new blockchain-based electronic stamp duty in the system we call `Smart Stamp Duty (SSD)`. The proposed system aims to expand the tax base for the stamp duty as it enables the recording of electronic documents' stamp duty payments. The benefit of our solution is threefold:

1. to increase the convenience for the taxpayers to pay the stamp duty,
2. to increase the state revenue in the stamp duty sector, and
3. to increase the transparency and auditability of the stamp duty mechanisms.

Our solution also enables the format conversion of the document from electronic format into paper-based format where the integrity of both documents can be verified.

For future work, we plan to evaluate the privacy issue of our proposed smart contract system. It is possible that the users' privacy is analysed based on the pattern of the transactions, thus expose risks on their privacy. We will develop mitigation strategies based on our analysis and implement them on the next version of our design. We also plan to evaluate the privacy issues of our system. Although the users will not put their identities on the blockchain, empirical analyses can be conducted to evaluate and discover the users, including their behaviors, locations, or even their real identities.

References

1. Ainsworth, R.T., Alwohaibi, M.: Blockchain, bitcoin, and VAT in the GCC: The missing trader example (2017)
2. Ainsworth, R.T., Alwohaibi, M., Cheetham, M.: The GCC's cryptotaxcurrency, VATCoin (2016)
3. Ainsworth, R.T., Shact, A.: Blockchain (distributed ledger technology) Solves VAT Fraud (2016)
4. Ainsworth, R.T., Viitasaari, V.: Payroll tax & the blockchain (2017)
5. Alaudin, A.: Formulation of stamp duty regulations on e-commerce transactions in Indonesia. *Journal of Faculty of Law, Brawijaya University* (2016)
6. Bloom, B.H.: Space/time trade-offs in hash coding with allowable errors. *Commun. ACM* **13**(7), 422–426 (1970)
7. Darono, A.: Study of the electronic stamp duty business model, April 2013
8. Megawati, A.: The police arrested the seller of thousands of fake stamps (2018). <https://www.merdeka.com/jakarta/polisi-bekuk-penjual-ribuan-meterai-palsu.html>
9. Republic of Indonesia Ministry of Finance. Book ii financial note and the draft 2018 state revenue and expenditure budget (2018). <http://www.anggaran.kemenkeu.go.id/content/Publikasi/NK%20APBN/2018%20Buku%20II%20Nota%20Keuangan%20Beserta%20RAPBN%20TA%202019.pdf>
10. Republic of Indonesia Ministry of Finance. Information of state revenue and expenditure budget (2018). <https://www.kemenkeu.go.id/media/6552/informasi-apbn-2018.pdf>
11. Republic of Indonesia Ministry of Finance. State revenue and expenditure budget (2018). <https://www.kemenkeu.go.id/apbn2018>
12. Nakamoto, S.: Bitcoin: A Peer-to-peer Electronic Cash System (2008). <http://bitcoin.org/bitcoin.pdf>
13. Republic of Indonesia. Law of the republic of indonesia number 13 of 1985 concerning stamp duty (1985). <https://jdih.kemenkeu.go.id/fulltext/1985/13tahun~1985UU.HTM>
14. Republic of Indonesia. Law on transactions and electronic information (2008). <https://jdih.kemenkeu.go.id/fulltext/2008/11tahun2008uu.htm>

15. Republic of Indonesia. Law of the republic of indonesia number 15 of 2017 concerning the state budget for budget year (2018). <https://jdih.kemenkeu.go.id/fullText/2017/15TAHUN2017UU.pdf>
16. Directorate General of Taxes. Director general of taxes decree number kep-122d/pj./2000 concerning procedure for repayment of stamp duty by affixing customs signs with a computerized system (2000). <https://www.ortax.org/ortax/?mod=aturan&page=show&id=1151>
17. Salebu, J.B.: Exploring potential revenues to reach tax targets (2018). <http://www.pajak.go.id/article/menggali-potensi-penerimaan-untuk-capai-target-pajak-2018>
18. Wijaya, D.A., Darmawan, O.: Blockchain from Bitcoin for the World. Jasakom, Jakarta (2017)
19. Wijaya, D.A., Liu, J.K., Suwarsono, D.A., Zhang, P.: A new blockchain-based value-added tax system. In: Okamoto, T., Yu, Y., Au, M.H., Li, Y. (eds.) ProvSec 2017. LNCS, vol. 10592, pp. 471–486. Springer, Cham (2017). https://doi.org/10.1007/978-3-319-68637-0_28
20. Wood, G.: Ethereum: a secure decentralised generalised transaction ledger. Ethereum project yellow paper **151**, 1–32 (2014)
21. Yanwar, H.J., Jati, W.: Extensification of stamp duty on electronic documents (2018). <http://www.pajak.go.id/article/menggali-potensi-penerimaan-untuk-capai-target-pajak-2018>