Decentralized Trustworthiness Score Management with Smart Contracts on the TrustLend Platform

Wisnu Uriawan[1,3][a], Youakim Badr[2][b], Omar Hasan[1][c] and Lionel Brunie[1]
[1]*Institut National des Sciences Appliquées de Lyon, 20 Avenue Albert Einstein, Villeurbanne CEDEX, France*
[2]*The Pennsylvania State University, Malvern, PA, USA*
[3]*Department of Informatics, UIN Sunan Gunung Djati, Jl. A.H. Nasution No.105 Bandung, Indonesia*
{*wisnu.uriawan, omar.hasan, lionel.brunie*}*@insa-lyon.fr, yb@psu.edu*

Keywords: Blockchain, Ethereum, Lending Platform, P2P, Scalable, Trustworthiness score.

Abstract: The personal lending marketplace, known as Peer-to-Peer (P2P) lending, has increased globally. However, providing unsecured loans to peers without requiring collateral remains a challenge. We propose a platform called TrustLend to enable trustworthy transactions in the personal lending application. The platform attempts to eliminate or minimize the collateral requirement. The trustworthiness score adds to this platform's variable selection rules and can help lenders decide on reliable candidates as borrowers. We also describe the prototype implementing the TrustLend platform based on Ethereum smart contracts that use the trustworthiness score and illustrate it with a Decentralized Application (DApp) case study and customized smart contracts. The prototype demonstrates fundamental features and supports borrowers, lenders, and recommenders in establishing proposals and approvals. Finally, the prototype shows how end-users can easily access loans with reduced collateral without hidden costs and swift transactions.

1 INTRODUCTION

In general, minor and micro-businesses and individual debtors find it difficult to get loans from banks without access to loan guarantors and collateral (Pokorná and Sponer, 2016; Uriawan. et al., 2021). In P2P lending, borrowers directly interact with peer lenders, making financing more accessible so that they can get financing more efficiently (Ma et al., 2018; Mammadli, 2016; Orús et al., 2019), which means a higher credit risk for lenders. Credit risk is the possible loss a bank or other lender suffers after offering a loan to a borrower. They include the actual risk of the borrower defaulting on the loan on time and the potential risk of default due to a decrease in credit score (Liang et al., 2017; Ma et al., 2018; Mammadli, 2016; Orús et al., 2019) or a reduction in the borrowers' ability to repay, and the lending platform getting profitable (Ma et al., 2017; Nizar et al., 2017; Tang, 2019).

P2P lending continues to increase worldwide every year. For example, in 2013, it reached 3.5 billion U.S. dollars. In the United States, 26 percent of people have used P2P payments for everyday purposes. In 2018, the value of mobile P2P payments reached 86 billion U.S. dollars and will continue to increase until now. P2P lending is a new trend in the "sharing economy". An exponential increase is estimated to reach one trillion U.S. dollars in 2050 (Statista, 2015; Uriawan. et al., 2021). However, a P2P lending platform can also create risks for lenders when the borrower cannot make payments according to the agreement. Trustworthiness (Arya et al., 2013; Tunç, 2019; Uriawan. et al., 2021; Xia et al., 2018) is a critical component in deciding for lenders whether borrowers are accepted or rejected to get some loans. However, in the end, it burdens borrowers in terms of interest and administrative costs. The bank or financial institutions have taken many borrower assets due to not fulfilling payments or experiencing delays in payments.

Blockchain technology is emerging and successfully applied in many business applications, such as banking and other financial institutions (Lee and Shin, 2018; Kiviat, 2015; Xie et al., 2020). Blockchain technology encourages our motivation to study the potential of the Ethereum blockchain (Coblenz, 2017; Norta and Leiding, 2019; Thakre et al., 2020). Re-

[a] https://orcid.org/0000-0001-6922-4705
[b] https://orcid.org/0000-0002-8976-7894
[c] https://orcid.org/0000-0003-1717-3867

cently, Blockchain technology has been applied to P2P and crowdfunding lending systems (Assessment and Smart, 2019; Chen and Micali, 2019; Dannen, 2017). The benefit of this new technology has led to explosive growth in the blockchain-based application, which exists within a highly secure system. Distributed ledger technology allows transaction and problem settlement without third-party risk (Millard, 2018; Taylor et al., 2020). Substantial work has also been done on using smart contracts for the monetization of data (Suliman et al., 2019), decentralization of management and governance (Zhang et al., 2022), and the improvement of reliability and security (Karamitsos et al., 2018; Karamitsos et al., 2018; Lombardi et al., 2018) in various systems.

The access to credit provided by the personal lending platform is intended to let the world of blockchains grow beyond the economic limitations of simply traditional money transactions. Loans (Capital, 2021) are not only an important economic factor, but also a vital component of personal financial freedom and give individuals greater purchasing power. Like those accessible through the personal lending platform, products offer a revolution in personal finance by granting (Battah et al., 2020) control over the medium of exchange to lenders of blockchain-based applications who wish to help the people and grow assets rather than spend them (Zhang et al., 2018).

1.1 Contributions

This paper makes the following major contributions:

- This paper introduces TrustLend as a personal lending platform and presents its fully functional prototype design and implementation details.
- The prototype shows the features required by borrowers, recommenders, and lenders to enable trustworthiness by implementing trustworthiness scores with Ethereum smart contracts.
- We describe the prototype architecture and conduct experiments and various personal loan simulations.
- This paper is an extended version of the conference paper published in SECRYPT 2022 (Uriawan. et al., 2022). This paper extends the conference paper by more than 30% and provides a much more in-depth discussion of the topic.

1.2 Outline

This paper's remainder is structured as follows: Section 1 introduces the potential of a personal lending platform, section 2 related work. Section 3 is our proposal for the trustworthiness prototype for a personal lending platform. Section 4 implements of the prototype for a personal lending platform, functional and non-functional requirements, input/output design, and implementation. Section 5 is discussion of our research. Section 6 concludes and future work of this paper.

2 RELATED WORK

This section evaluates the lending platforms and in particular we observe their advantages and disadvantages. The parameters of lending platform services include registration required, interest rate, lending or borrowing provided, using own token, and collateral required. These parameters correspond with several existing lending platform features as shown in Table 1.

2.1 Everex

Everex is a financial technology that creates decentralized, global credit histories and scorings for individuals and SMEs. The Everex supports transfers, borrowing/lending, and trading in any fiat currency from anywhere in the world. Everex uses Crypto cash Ethereum ERC20 tokens and as well as fiat currencies. The Everex is implemented on the Ethereum blockchain and uses Solidity as a smart contract language. Similar to the traditional lending system, the Everex environment still involves fiat money and requires collateral (Norta and Leiding, 2019). In contrast to Everex, the TrustLend platform proposed in this work does not involve fiat currencies and does not require collateral.

2.2 ETHLend

ETHLend is an Ethereum-based decentralized lending platform connecting borrowers and lenders. It allows anyone to lend or borrow with an Ethereum address. ETHLend is decentralized lending on the Ethereum network using ERC-20 compatible tokens or Ethereum Name Service (ENS) domains as collateral. ETHLend reduces the loss of loan capital on default. Some features allow for the marketplace to manage, such as interest rate and collateral value. However, the pseudo-anonymous nature of the Ethereum blockchain network opens the possibility to avoid repayment of the loan since the lender might not have all the necessary details of the borrower to enforce the debt in the borrower's jurisdiction (Network

et al., 2018).

The TrustLend is applying the trustworthiness score to help the lenders/investors to identify the eligible borrowers to minimize the loss. We have an interest rate formula with Annual Percentage Yield (APY) as the standard interest rate that is possible to apply in a blockchain lending platform considering the effect of compounding interest and does not require collateral as well.

2.3 WeTrust

WeTrust is an Ethereum blockchain that provides mutual aid on equal footing to borrowers with existing social capital and trust networks. Trusted Lending Circles are proposed to create a Rotating Savings and Credit Association (ROSCA) powered by smart contracts. It eliminates the need for a trusted third party, which cuts fees, improves incentive structures, and decentralizes risks. It will eventually incorporate mutual insurance, voting within reciprocal aid organizations, and P2P lending. However, WeTrust may not be suitable for personal users because it still requires collateral (Token, 2018). Our TrustLend platform offers to focus on the P2P lending platform to help people get loans without collateral.

In summary, we have an opportunity to propose the TrustLend platform with the advantages such as no collateral required, using social recommendations, and cultivating borrowers' trustworthiness score.

3 OUR PROPOSAL

This section describes the prototype architecture, trustworthiness formula, and TrustLend Prototype development principles.

3.1 TrustLend Architecture

The architecture shows a DApp platform (Uriawan et al., 2021) for Ethereum blockchain-based personal lending to assist borrowers, recommenders, and lenders/investors in the lending process. This architecture minimizes or eliminates the collateral need by assessing the borrower's trustworthiness. The users who interact with the system as borrowers, recommenders, and lenders/investors are shown in Figure 1. Smart contracts provide the functionality of the trustworthiness scores, recommendations, and Metamask wallet. The borrowers', recommenders', and lenders'/investors' transactions will be stored on the Ethereum blockchain.

3.2 Trustworthiness Score

The trustworthiness score formula (Uriawan. et al., 2021) estimation is based on the user behavior attributes of risky attitude, trustworthiness, time preference, and impulsiveness (Arya et al., 2013). It can be used to determine the correlation of the behavior as reflected in the credit score. The standards of impatience, trustworthiness, and impulsivity affected credit scores (Tunç, 2019). We adopt the trustworthiness score formula in terms of the reliable borrowers in one online prototype, as follows Equation 3 and Equation 4:

$$\begin{aligned} &Trustworthiness_{Score} \\ &= Profile_{score} + Activity_{score} \\ &+ Social\ Recommendation_{score} \\ &+ LoanRisk_{score} \end{aligned} \tag{1}$$

And we added positive weight for each variable, as follows:

$$\begin{aligned} &Trustworthiness_{score} \\ &= W_p * Profile_{score} + W_a * Activity_{score} \\ &+ W_r * SocialRecommendation_{score} \\ &+ W_l * LoanRisk_{score} \end{aligned} \tag{2}$$

Where:
Trustworthiness_score: Borrower credit score.
Profile_score: Personal information of Borrower.
Activity_score: Business activity or job information of Borrower.
SocialRecommendation_score: The recommendation value of Borrowers from Recommender.
LoanRisk_score: Information of the record from another loan of Borrower.

3.3 Trustworthiness Score Splitting Formula

We briefly discuss membership functions variables rules for making a decision and define the trustworthiness score in terms of four variables (Uriawan. et al., 2021), namely *LAPS (Loan Risk, Activity, Profile, and Social Recommendation)* as borrower trustworthiness score (Martens et al., 2007; Uriawan et al., 2022), see Equation 1. The authors applied the Bank Marketing dataset from UCI public dataset (Moro et al., 2014) show in Fig. 9:

1. *Loan Risk score* is the component for measure the borrower candidate has the other loan such as housing, car, etc. in Fig. 2 (a), (b), if there is any

Table 1: Ethereum Lending platforms Evaluation (Tran, 2019)

Lending Platform	Registration Required	Interest Rate for Loans (Min.)	Lend or Borrow	Own Token	Required Collateral
Everex	Yes	Market	Both	Yes	Yes
ETHLend	Yes	Market	Both	Yes	Yes
WeTrust	Yes	Market	Both	Yes	Yes
TrustLend	No	APY	Both	Yes	No

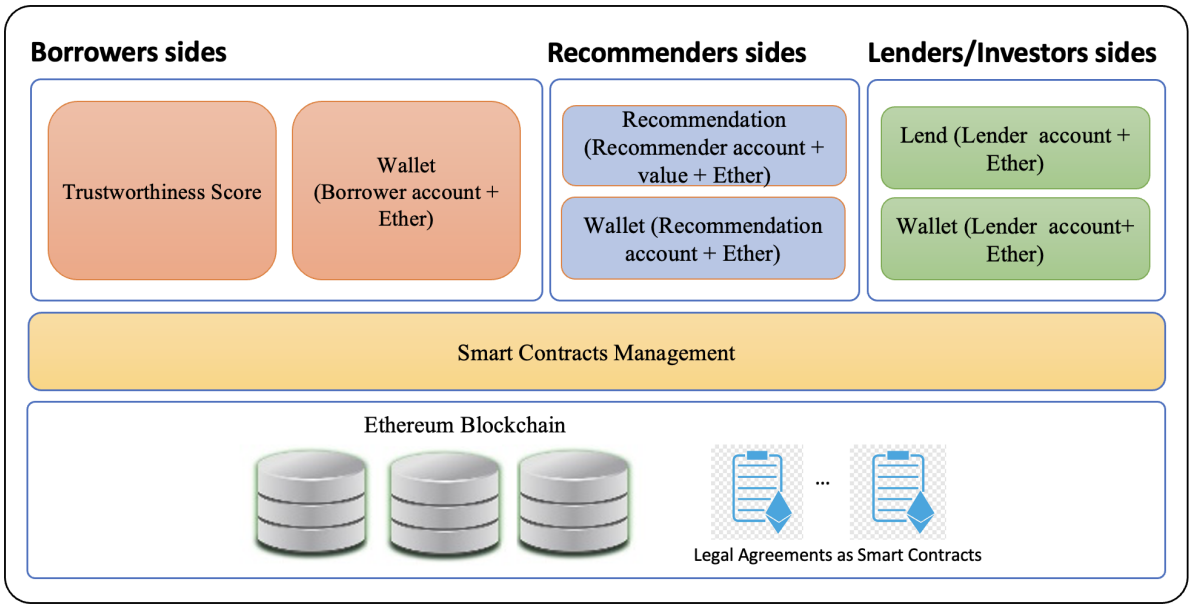


Figure 1: TrustLend Architecture Design.

another loan is risky to allowing get another loan and will decreasing the trustworthiness score, see Equation 3.

$$Loan\ Risk\ score = \sum_{i=1}^{n}(w_i * L_i) \tag{3}$$

where:
w = Weight for each variable $\{w\ in\ \mathbb{R} \mid w \leq 1\}$, that able to be defined by user.
i = Sequence of weight and variable.
L = Variables (loan, housing), where $\{L\ in\ \mathbb{Z} \mid L \leq 100\}$, and scale of values are between 0 to 100.

2. *Activity score* describing the borrower activity in occupation such as job or business activity in Fig. 3 (a), (b), to measure the ability to pay and considering the credit plafond or credit limit that correspond with their activity, if borrower candidate has a good occupation they will get the highest value of *Activity score*, see Equation 4.

$$Activity\ score = \sum(A) \tag{4}$$

where:

```
print(df[['loan','housing']])
```

	loan	housing
0	no	yes
1	no	no
2	no	yes
3	unknown	unknown
4	no	yes
...	...	...
4114	yes	yes
4115	no	yes
4116	no	no
4117	no	no
4118	no	yes

[4119 rows x 2 columns]

```
print(df.loan.unique())
['no' 'unknown' 'yes']

print(df.housing.unique())
['yes' 'no' 'unknown']
```

(a) (b)

Figure 2: Loan and Housing Dataset (a) and List of Loan and Housing (b)

A = Variable (Job activity), where $\{A\ in\ \mathbb{Z} \mid A \leq 100\}$, and scale of values are between 0 to 100.

3. *Profile score* is the personal data of borrower candidates such as age, education level, and marital status in Fig. 6. These variables support to trustworthiness score. For example, the borrower should be older than 18 years old and 88 years old maximum age show in Fig. 4 (a), (b), (Kellison and London, 2011), have an education level in Fig. 5 (a) to measure the economy and activity

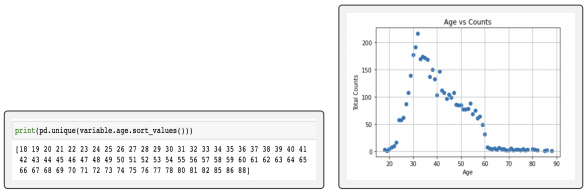
(a) (b)

Figure 3: Job Dataset (a) and List of Job (b)

in industry or entrepreneur, and have marital in Fig. 5 (b) status to consider the family dependent. All the variables summarise in one variable as *Profile score*. The formula to get the *Profile score* is shown in Equation 5:

$$Profile\ score = \sum_{i=1}^{n} (w_i * P_i) \tag{5}$$

where:
w = Weight for each variable $\{w\ in\ \mathbb{R} \mid w \leq 1\}$, that able to be defined by user.
i = Sequence of weight and variable.
P = Variables (age, education, marital) are $\{P\ in\ \mathbb{Z} \mid P \leq 100\}$, where scale of values is between 0 to 100.



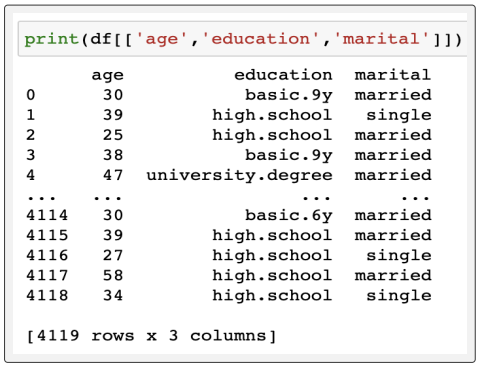
(a) (b)

Figure 4: Range of Age (a) and Diffusion of Age (b)

(a) (b)

Figure 5: List of Education level (a) and List of Marital Status (b)

4. *The social recommendation score* is the primary variable the borrower gets support directly from the other users to add recommendation value. This values to as a guarantor for borrowers to get some loan from lenders/investors through the lending platform see in Equation 1, 2. *Social Recommendation score* = variables S (Social Recommendation) are $\{S\ in\ \mathbb{Z} \mid S \leq 100\}$, where scale of values is between 0 to 100.



```
print(df[['age','education','marital']])

      age           education  marital
0      30           basic.9y  married
1      39         high.school   single
2      25         high.school  married
3      38           basic.9y  married
4      47  university.degree  married
...   ...                 ...      ...
4114   30           basic.6y  married
4115   39         high.school  married
4116   27         high.school   single
4117   58         high.school  married
4118   34         high.school   single

[4119 rows x 3 columns]
```

Figure 6: A Profile Dataset

With $\{W\ in\ \mathbb{R} \mid W \leq 1\}$, and W_p, W_a, W_r ,*and* W_l are positive weights of the trustworthiness parameters such that $W_p + W_a + W_r + W_l = 1$. The weights of the trustworthiness attributes are predetermined based on their priority value that can modify by consensus. For example, $W_p = 0.25$, $W_a = 0.2$, $W_r = 0.3$ $W_l = 0.25$. In this example, social recommendation is given the highest value whereas activity is given the lowest value it's show that the social recommendation is the priority to measure the good borrower candidate.

The trustworthiness score we propose is a value of borrowers set by the smart contract so that both parties understand each other's obligations and risks that will be accepted. The variables include profile, activity, social recommendation, and loan risk, as shown in Equation 1. The borrowers propose some loans with their trustworthiness score, determining whether the lenders/investors are able to grant the loan. Trustworthiness scores value will increase, and the borrower can get a more significant opportunity in the next submission.

The system will reduce the trustworthiness score if the borrower receives a bad report from lenders/investors or recommenders. The borrower will get a high trustworthiness score, making it easier to get the loan in the next cycle. The smart contracts management handles the borrowers, recommenders, and lenders/investors from the available services on the Ethereum blockchain.

3.4 TrustLend Prototype Development Principles

The prototype principles we adopt are standards coding and conventions, automated unit testing, and static analysis tools. Some regulations relate to our prototype, as follows (Brown, 2013; Uriawan. et al., 2021):

1. **Layering strategy**, a layered architecture usually results in a software system with a high degree of flexibility because each layer is isolated

from those around it. The prototype applies a layers strategy to make every design flexible for all users (borrowers, recommenders, and lenders/investors).

2. **Placement of business logic**, our prototype ensures that business logic permanently resides in a single place for reasons related to performance or maintainability among stakeholders (borrowers, recommenders, and lenders/investors).
3. **High cohesion and low coupling**, our prototype focuses on building small, highly cohesive blocks that do not require too many dependencies to do their job part-by-part development related to our architecture design of the prototype.
4. **Use of the HTTP session**, the HTTP session for storing temporary information between requests. The prototype can often depend on many things, including scaling strategy, where session-backed objects are stored, what happens in the event of a server failure, whether using sticky sessions, and the overhead of session replication.
5. **Always consistent versus Eventually consistent**, prototypes have discovered that it often needs to make trade-offs data to meet complex non-functional requirements, such as updating information is required as fast as possible (e.g., 24/7 services). In this context, consistency is appropriate, but a consistent approach is fundamental.

4 IMPLEMENTATION

The trustworthiness personal lending platform prototype is a client-blockchain serverless application, where the entire flow of the app happens between the client and the blockchain. The client code can be hosted anywhere, and AWS (Amazon Web Services) with S3 (Simple Storage Service) features, Google Cloud, Github Pages, Netlify, other cloud providers, or your own server. Our prototype is able to query the blockchain, and we use a web3 provider Metamask. A browser extension (available for Chrome and Firefox) handles the actual web3 connection to a node shown in Figure 7.

For instance, all the business logic, loans, and user history are handled and stored in the blockchain, which is decentralized. But since the Ethereum blockchain platform (or any other EVM (Ethereum Virtual Machines) (Modi, 2018) blockchain-based like Polygon). Charges fees for each written transaction. It is ubiquitous to store the data not used in smart contracts calculations elsewhere to pay fewer fees. We have chosen the IPFS (Interplanetary File System) (Sicilia et al., 2019) to store the loan description, images, and necessary data supported. Once the data is stored in the IPFS, the content identifier (CID) is returned and stored in the loan smart contracts to find this data later. We use NFT (Non-Fungible Token) (Buterin, 2014) storage (Free, decentralized storage and bandwidth for NFTs) to store the project's Info into IPFS.

4.1 Smart Contracts

The main smart contract that the client interacts with is the loan controller. It creates loans, handles investments, recommendations, repayments, etc. From the moment the user applies for a loan, the apply for loan function in the loan controller is called and creates a unique loan contract related to the loan in question. The smart contracts necessary of information about the loan:

1. Borrower (represented by User contract instance)
2. Requested amount
3. Repayment's count
4. Interest
5. Loan creation date
6. Last repayment date
7. Return the amount
8. Lenders/Investors(array)
9. Recommenders(array)
10. Tscorecontroller contract (to handle user's trustworthiness score).

The recommenders and lenders/investors can call functions in the LoanController to lend/invest and recommend by providing the address of the LoanContract. These smart contracts require a communication process and are defined as a legal agreement between borrowers, recommenders, and lenders/investors, is shown in Figure 8.

4.2 Implementation of the Trustworthiness Score Formula

The implementation of the trustworthiness score formula is shown in Figure 9. We used the Bank Marketing subset from the UCI public dataset (Moro et al., 2014). The borrowers get the value of the loan risk, activity, and profile score by converting the value of the dataset. The social recommendation score will be obtained from the other user as recommenders. The prototype will count each variable to become a trustworthiness score (Tscore), which is used as the final decision from lenders/investors to grant the loan.

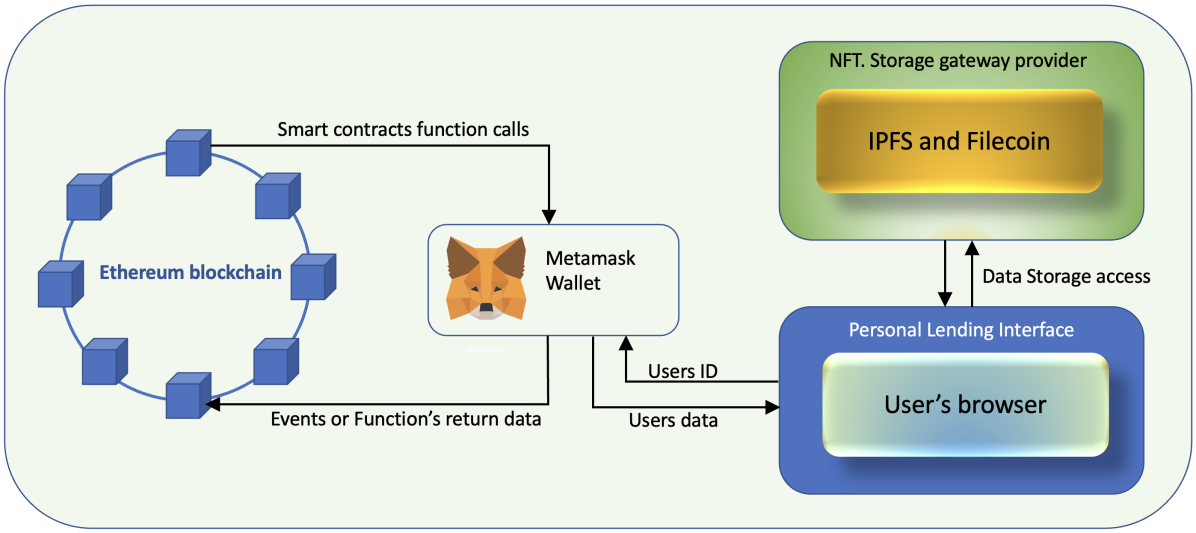


Figure 7: Prototype of a Trustworthy Personal Lending Platform.

The borrowers' trustworthiness score present is a commitment between borrowers, recommenders, and lenders/investors managed by the smart contract. The objective is for all parties to understand the obligations and its risk. All variables data (loan risk, activity, profile, and social recommendation score) will be assessed as the borrowers' trustworthiness score. After the borrowers get the score, they can propose a loan and get the loan corresponding to their score. The borrowers' trustworthiness scores will increase alongside the record of the borrowers' installment payments.

4.3 The ERC-20 Token Standard

The Ethereum Request-for-Comments #20 is called ERC-20 standard token, which possibly allows for fungible tokens on the Ethereum blockchain. The standard provides functions that include the exchange of tokens among accounts, such as getting the current token balance of an account and the total supply of the token available on the network.

ERC-20 Token Contracts are the smart contracts correctly implemented and keep the records of created tokens on Ethereum. An Ethereum introduced a complete programming language to write and execute smart contracts in decentralized applications on the Ethereum Virtual Machine (EVM) and Ethereum blockchain. Smart contracts are autonomous, immutable, conducted in the EVM manner, and stored in the Ethereum blockchain. The smart contracts are able to hold ETH in ERC-20 tokens (Andesta et al., 2020; Ethereum.org, 2022).

On the other hand, the standards application level for creating tokens, naming, and library register. The ERC-20 tokens became the first standard of crowdfunding, and its applications for Ethereum blockchain-based decentralized applications are able to reduce the complexity. The ERC-20 token standard is applied in our prototype and is able to implement in localhost with ERC20 token based and the testnet KOVANETH running on Optimism Testnet Network (Andesta et al., 2020; Ethereum.org, 2022).

4.4 Vulnerability and Countermeasures

The Ethereum blockchain has typical vulnerabilities, including scalability, forking, and security. There are several security vulnerabilities in Ethereum blockchain-based smart contracts, which sometimes do not behave as intended when the users change the network. Limitations of the smart contracts file size impact the DApp performance as well.

Loan default is one of the issues that may be faced by the TrustLend platform as with any other lending platform. TrustLend implements some conditions for all users (borrowers, recommenders, and lenders/investors) in order to prevent losses of the lenders/investors. To minimize the loss, each borrower may be required to obtain a certain minimum number of social recommendation. The defaulting borrowers' accounts may also be banned after some installments are missed.

Another issue is that the borrower and the recommender could be the same individual with different accounts. However, to make sure that identical users are not able to cheat on the platform, TrustLend will identify the users (borrowers, recommenders,

Tscore Smart Contracts (SocialRecommendations)

```
pragma solidity ^0.8.0;

import "./TScoreControllerInterface.sol";
import "./User.sol";
import "./Loan.sol";

contract    TScoreController    is    TScoreControllerInterface,
ExponentialNoError {

constructor() {
        admin = msg.sender;
    }

function updateSocialRecommendationScore(address _loanAddr,
address _userAddr, uint _recommenderWeight, uint _givenScore)
external override returns (uint) {
        User user = User(_userAddr);
        Loan loan = Loan(_loanAddr);
        Exp memory weightedScore = Exp({mantissa: 0});
        Exp[] memory weights;
        uint8[] memory scores;

(weights, scores) = loan.getRecommendersWeightsAndValues();
        for (uint i = 0; i < weights.length; i++) {
            weightedScore = add_(mul_(weights[i],
scores[i]), weightedScore);
        }

user.setSocialRecommendationScore(weightedScore);
        return user.getTScore();
    }

function updateActivityScore(address _loanAddr, address
_userAddr, uint _recommenderWeight, uint _givenScore) external
override returns (uint) {
        User user = User(_userAddr);
        return user.getTScore();
    }

function updateProfileScore(address _loanAddr, address
_userAddr, uint _recommenderWeight, uint _givenScore)  external
override returns (uint) {
        User user = User(_userAddr);
        return user.getTScore();
    }

function updateLoanRiskScore(address _loanAddr, address
_userAddr, uint _recommenderWeight, uint _givenScore) external
override returns (uint) {
        User user = User(_userAddr);
        return user.getTScore();
    }

function updateTScore(address _loanAddr, address _userAddr, uint
_recommenderWeight, uint _givenScore) external override returns
(uint) {
        User user = User(_userAddr);
        return user.getTScore();
    }

function getTScore(address _userAddr) external view override
returns (uint) {
        User user = User(_userAddr);
        return user.getTScore();
    }
}
```

Figure 8: Smart Contracts Trustworthiness score.

and lenders/investors) with their Metamask wallet address. This will ensure that there are no similar address accesses with different roles for the same loan. For example, the borrower cannot become the recommender or lenders/investor when the borrowers propose a loan, and both as well. However, we note that this countermeasure is not foolproof at all and a sophisticated attacker may bypass detection. We hope to strengthen the countermeasures of TrustLend in future work in order to improve security against this attack.

4.5 Lending Transaction Process

The app's client is built on React framework, an open-source javascript library. The app has the following main screens for the user interface:

4.5.1 The Main page

The main page describes the main menu of the prototype. It is provided for borrowers, recommenders, and lenders/investors. Users can access the specific menu after being connected to the Metamask wallet. The prototype combines trustworthiness and consensus in a legal agreement between borrowers, recommenders, and lenders/investors.

Therefore, representing the blockchain's data, processes, and transactions is required. Users can obtain permission only through the prototype and integrating components, such as security, speed, immutability, scalability, and resilience, including ledgers that can be changed through only the consensus. Metamask wallet is required by prototype, and users are able to install individual with terms and conditions, is shown in Figure 10.

The prototyping functionality is offered to three users: Borrowers, Recommenders, and Lenders/Investors. The borrower can access the menu on the borrower page. Before accessing the prototype, they (Borrowers, Recommenders, and Lenders/Investors) should have the Metamask wallet. The borrower login first to their wallet. After the loan application has been received, the borrower user is able to withdraw the loan and make installment payments according to the agreements.

The lenders/investors user are able to access their menu and invest with a selection of borrowers who propose the loan. In these cases, the lenders/investors users determine the allocation of funds desired. The recommenders' users can access the prototype menu to give a recommendation score to the borrowers and the ETH values. The lenders/investors can use the trustworthiness score to decide and grant the loan.

Users manage the private key to receive the payments per transaction by their wallets. For other transaction payments, unsigned transactions are sent from the wallet to the prototype, verified by the users (borrowers, recommenders, and lenders/investors) on the personal wallet screen, confirmed via Metamask as third-party, then signed by the user ID, and sent back to the users' wallet.

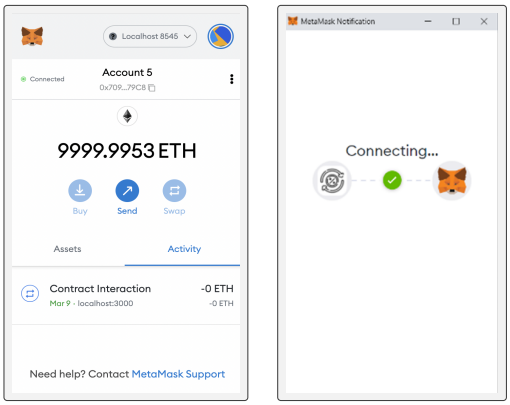
4.5.2 Borrower page

The borrowers is shown in Figure 11 can access the prototype; the system provides how the borrowers propose some loans and terms and conditions. Some borrowers' users give some loan information as new borrowers and signals are sent to all lenders. The borrower page is provided to borrowers when trying to

Platform users

Profile ID	Borrower contract address	Borrower wallet address	Total TScore	Activity score	Profile score	Social recommendation score	Loan risk score	# cycles
0	0x47b36E824631e97E94d462F822181Ed08A459384	0x8626f6940E2eb28930eFb4CeF49B2d1F2C9C1199	71	65	60	61	100	1
1	0x46BCe031ea52F0A6B2096cc3abFA0CFA0cAF06C7	0x70997970C51812dc3A010C7d01b50e0d17dc79C8	75	78	80	66	79	1
2	0x387dD5D71157554397507 51FC78578416660f455	0x06f509F73eefBA36352Bc8228F9112C3786100dA	54	56	58	50	57	1
3	0x8f46f37Ec07D3f52A3D93720793D48E5456C246B	0x35CbD98014D4C3825d8486993b9944c7Be56f763	70	78	80	50	79	1
4	0x7625092439918977e9573653C6AF7275CC3BF25b	0x4cFF48a294272A0449d9163E92E6c93ba0CD90B5	54	56	58	50	57	1
5	0x142AB05a3fD31663ea627DB1698639D27061010C	0xe37AA5C872eC6987E0E899E0fFE8C74dAc99ff21	68	65	60	50	100	1
6	0x448909a30544E601E44F51bA728e7C5f08627C55	0x0eC5F4cD22a4EEF3fdC63a31c5b6A2418D429193	70	78	80	50	79	1
7	0x8744b336FC44395Da72850AD9f6220CF6Ea34962	0x64047426729dceF24d6bD58329bba0BAA95Ba3b0	54	56	58	50	57	1
8	0xbE114A1cD55A685C772E3Ec3CBB3e3afbE9b018F	0x1Df10644987013a9536f17092DbEa647f98f2e2B	68	65	60	50	100	1
9	0x6858aBC1D6AFCf3F9B78b0Bd33253F622EF4eC7d	0x343B14Ebb982be358cE495eCC7a018247794A945	70	78	80	50	79	1
10	0x0f280b6190204A51Cd06419c6f0F90E815FEc7f6	0x71d13E61AC7dC6773EB1A17b7f03aB13C38664EE	68	65	60	50	100	1
11	0x6FebE66a2Ca2cd21199d201e5a843eEf635C40fB	0x2Dde3b23908D5f027806A99F9ac77079EEB55032	70	78	80	50	79	1
12	0xd00c64B12Cd24D4DD97cE894575f19F4A092a859	0x67E28cc4C005027f5e2Ef1034B228f47C7e5c01F	54	56	58	50	57	1
13	0x23F2a9441B0eb4c3e1999C377f74bCe3BffC7008	0xEb9d54Ba366750e0Eb161262a82B4c9412CF956b	68	65	60	50	100	1
14	0xC695D11C8702B7DD71f1Ccaa030A56bF212eE97B	0x7ADd4eFF69bBcB4666949742eC19428e9d4742aA	70	78	80	50	79	1
15	0x231673F8f289CB8AB09F1050920d5317C954dD53	0xa9B2c325C56d048AA1EB5804f89410903D2cDfd8	54	56	58	50	57	1
16	0xBe928d8E377719681B13f5bC34fc6930d0f0aE57	0x13f3FdD8A4d1bAdD65D1a241b9bBc86Bf7575a03	50	50	50	50	50	1

Figure 9: Implementation of the Trustworthiness Score Formula .



(a) (b)

Figure 10: Trustlend Metamask Wallet (a) and Metamask connecting process (b)

apply for some loans, with the proposed loan amount, installment period, and loan description being the purpose of the loan.

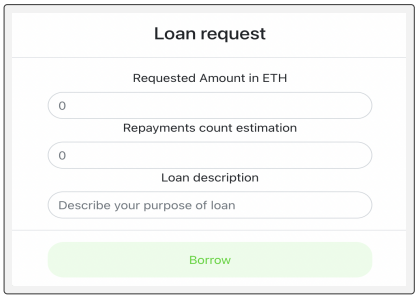
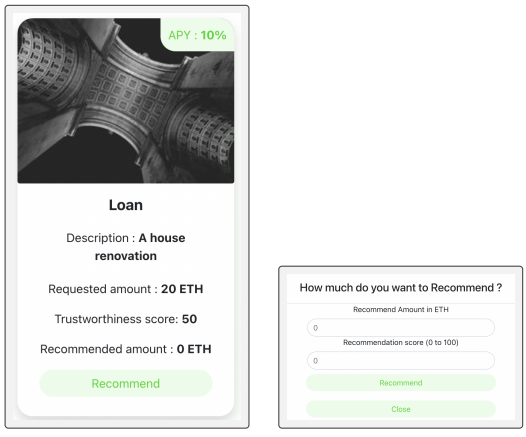


Figure 11: Borrower loan request.

4.5.3 Recommender page

The recommender can access the lending platform with their wallet. The prototype will provide the borrower who needs recommendations, then the recommender gives some ETH, and the score/value is shown in Figure 12. The prototype provides the recommendation page for making sure the lenders/investors are able to grant the borrower's loan proposal. On the other hand, the recommender can recommend to all the borrowers who request the loan, and then they can make a profit after borrowers finish their installments.

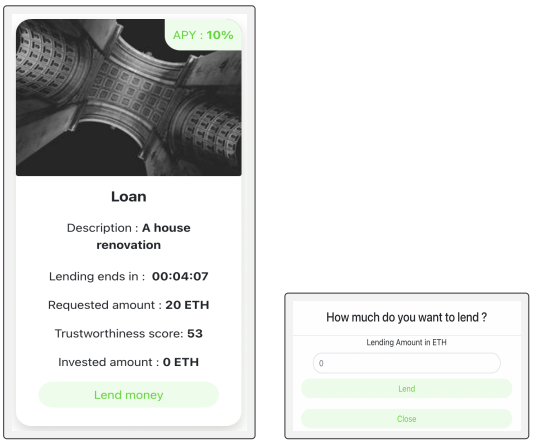


(a) (b)

Figure 12: The Social Recommendation information request (a) and the Social Recommendation input value (b)

4.5.4 Lender page

The lender/investor page is provided for lenders/investors to search the eligible borrowers. This page includes information on borrowers, loan amount, and interest in Annual Percentage Yield (APY). The prototype presents the borrowers who proposed a loan. The lenders/investors get an opportunity to choose the eligible borrowers to grant the loans based on the trustworthiness score, as shown in Figure 13.



(a) (b)

Figure 13: The Borrower information request (a) and Lender/Investor page input value (b)

The smart contracts as a legal agreement (Borrowers, Recommenders, and Lenders/Investors sides) are the core of the lending prototype we are proposing. The excellent trustworthiness score of borrowers is a significant factor in this prototype, reducing guarantee dependence replaced by social recommendation, and other variables are supported. Many lending platforms and banks still require a guarantee, which is burdensome to the borrowers, because not every borrower is able to provide it.

4.5.5 Summary of the Loan Request

The borrower's loan information is described in detail of loans proposed for each borrower, including the amount requested, the number of lenders/investors' information, number of recommenders, trustworthiness score (TScore), activity score, profile score, social recommendations score, and loan risk score.

The prototype informs the amount to return with interest is set by the system with the Annual Percentage Yield (APY) formula, and information about the installments corresponds with the repayment count proposed. The last information of these outputs is the loan status. The current status of the loan is in a "recommendation phase", which means the process in the recommendation phase. After the recommendation phase, the system sent a signal to lenders/investors to make a decision and will change the loan status to "withdrawal" when the lenders/investors grant the loan, then the borrowers are able to withdraw their loans.

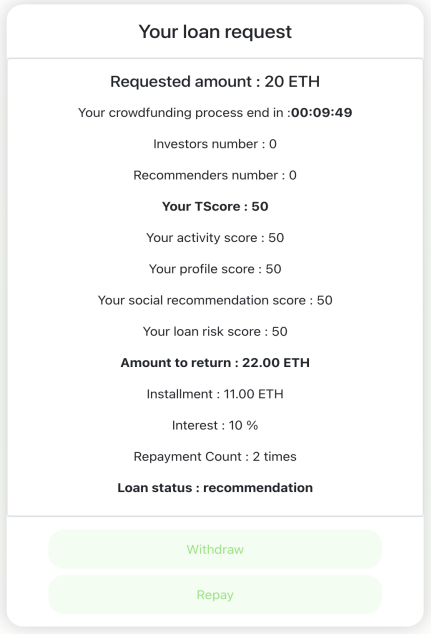


Figure 14: TrustLend Borrower Request a Loan.

It seems reasonable to expect a high credit score associated with the payments process if there is evidence of patience with current and future suitable consumptive activity with borrowing is shown in Figure 14.

5 DISCUSSION

The objectives of this prototype are to avoid impulsive borrowers who have difficulty resisting the temptation to borrow and increase debt for consumptive needs.

Lenders/Investors are able to monitor the borrower and manage their lend/investment by choosing eligible borrowers to minimize their losses. Each lender/investor can choose by determining borrowers who can pay off and get the highest trustworthiness score. The trustworthiness score formula is well defined (such as weight percentage, variables, etc.) before deployment to avoid mistakes.

5.1 Advantages

Blockchain technology has advantages with immutability, integrity, security, equal rights for all network members to get information, protect users' data from unauthorized access, and encryption. No personal information of borrowers, recommenders, and lenders/investors is shown in this prototype. We provide a prototype with an autonomous transactions process supported by smart contract functions after the deployment phase. Smart contracts pay attention to borrower trustworthiness scores on a personal lending platform, so lenders can consider the potential risks incurred.

The prototype is based on trustworthy personal lending that can provide a loan for borrowers who need it without collateral. The value of trust among borrowers, recommenders, and lenders/investors has a strong influence on a personal lending platform.

The social recommendation as a guarantor convinces the lenders/investors and minimizes the difficulty of granting the borrowers' loans. This prototype is one of the personal lending platforms suitable for P2P lending applications that apply blockchain advantages.

5.1.1 Incentives for Borrowers, Recommenders, and Lenders

The stakeholders (borrowers, recommenders, and lenders/investors) have the following incentives on the TrustLend platform:

- **Borrowers**: The borrowers do not have to provide collateral, which is a major incentive for them. Most lending platforms mentioned in Section 2 still require collateral. However, the borrowers are held accountable because that need to get good recommendations from social recommenders (the replacement for guarantors) and maintain a trustworthiness score. TrustLend keeps the record of the borrowers' trustworthiness score. The borrowers' have to show strong commitment and willingness to pay back the loan on time to maintain a good record.
- **Social Recommendations**: The recommenders are expected to give honest and frank recommendations about the borrowers because the recommendation scores will impact the trustworthiness score. The recommenders who correct recommendations will be incentivized as valuable recommenders and may gain benefit towards an increased trustworthiness score for the purpose of obtaining loans. Moreover, recommenders may also provide recommendations altruistically for the benefit of their fellow borrowers and lenders without the expectation of any incentives. TrustLend maintains the records of correct recommendations. When the borrowers pay the loans, the corresponding recommenders will get an incentive from interest profit sharing with the lenders/investors.
- **Lenders/Investors**: The lenders/investors grant the loan and choose the eligible borrowers themselves. Therefore, they know the risk of lending to a particular borrower and TrustLend provides the trustworthiness score for helping them choose the borrower and make the decision of lending to them. To the best of our knowledge, none of the lending platforms guaranteed the borrowers to pay back their loans, and this the case on the traditional lending systems as well. However, TrustLend maintains the borrowers' trustworthiness score in order for the lender to vet and identify good borrowers. They can also use the history of lending activity to make lenders/investors more confident.

  After the borrowers finish installments payments, the lenders/investors get the profit from the interest managed by TrustLend. TrustLend offers lenders/investors to invest their cryptocurrency as an alternative for investment and challenges with sharing profit interest rate using Annual Percentage Yield (APY) that is more progressive and competitive than the conventional approach. Since TrustLend makes loans easier due to the lack of collateral, we expect that the platform will attract a larger number of borrowers, which will also provide the lender of a broader group of borrowers to choose from.

5.2 Disadvantages

The disadvantage is that performing off prototype transactions will increase transaction time because of the need for recommendations and lender/investor approval. An increasing number of users can cause problems with scalability and processing times. The most straightforward approach is to increase the bandwidth capacity for borrowers, recommenders, and lenders/investors.

Therefore, the developer ensures all the smart contract functions are running well because it is not possible to change after deployment. In particular, all users know the risks and the borrower's trustworthiness.

6 CONCLUSION

The architecture design of the prototype of trustworthy blockchain-based personal lending can be concluded that the Ethereum blockchain can be used to create a personal loan to identify the potential of borrowers who are attractive to invest for lenders/investors. Also, we added a social recommendation to support the trustworthiness score component to convince the lenders/investors to grant the borrowers' loans.

This prototype is one of the lending platforms suitable for personal lending applications that apply the Ethereum blockchain advantages dimensions, such as anonymous, decentralized, immutability, and secure. This prototype proposes to minimize the difficulty by introducing smart contracts as a backbone application to support borrowers, recommenders, and lenders/investors. The prototype is expected to be implemented privately, in social network groups, and in small business environments that can be scalable to many members. They have advantages and disadvantages and are able to present in the future.

ACKNOWLEDGEMENTS

The first author wishes to acknowledge the MORA-LPDP (Ministry of Religious Affairs & Endowment Fund for Education of Indonesia) Scholarship from the Indonesian Government and INSA de Lyon LIRIS Laboratory UMR 5205 CNRS, which partially supports and funds this research work.

REFERENCES

Andesta, E., Faghih, F., and Fooladgar, M. (2020). Testing smart contracts gets smarter. In *2020 10th International Conference on Computer and Knowledge Engineering (ICCKE)*, pages 405–412. IEEE.

Arya, S., Eckel, C., and Wichman, C. (2013). Anatomy of the credit score. *Journal of Economic Behavior and Organization*, 95(47783):175–185.

Assessment, R. and Smart, U. (2019). Reputation Assessment Using Smart Contracts. pages 1–17.

Battah, A. A., Madine, M. M., Alzaabi, H., Yaqoob, I., Salah, K., and Jayaraman, R. (2020). Blockchain-based multi-party authorization for accessing ipfs encrypted data. *IEEE Access*, 8:196813–196825.

Brown, S. (2013). Software architecture for developers. *Coding the Architecture*.

Buterin, V. (2014). A next-generation smart contract and decentralized application platform. *Etherum*, January:1–36.

Capital, G. (2021). Rates, Terms & Speed of Funding.

Chen, J. and Micali, S. (2019). Algorand: A secure and efficient distributed ledger. *Theoretical Computer Science*, 777:155–183.

Coblenz, M. (2017). Obsidian: a safer blockchain programming language. In *2017 IEEE/ACM 39th international conference on software engineering companion (ICSE-C)*, pages 97–99. IEEE.

Dannen, C. (2017). *Introducing Ethereum and solidity*, volume 1. Springer.

Ethereum.org (2022). Erc-20 token standard.

Karamitsos, I., Papadaki, M., Al Barghuthi, N. B., et al. (2018). Design of the blockchain smart contract: A use case for real estate. *Journal of Information Security*, 9(03):177.

Kellison, S. and London, R. (2011). *Risk Models and Their Estimation*. ACTEX academic series. ACTEX Publications.

Kiviat, T. I. (2015). Beyond bitcoin: Issues in regulating blockchain tranactions. *Duke LJ*, 65:569.

Lee, I. and Shin, Y. J. (2018). Fintech: Ecosystem, business models, investment decisions, and challenges. *Business Horizons*, 61(1):35–46.

Liang, L.-W., Huang, B.-Y., Liao, C.-F., and Gao, Y.-T. (2017). The impact of smes' lending and credit guarantee on bank efficiency in south korea. *Review of development finance*, 7(2):134–141.

Lombardi, F., Aniello, L., De Angelis, S., Margheri, A., and Sassone, V. (2018). A blockchain-based infrastructure for reliable and cost-effective iot-aided smart grids.

Ma, B.-j., Zhou, Z.-l., and Hu, F.-y. (2017). Pricing mechanisms in the online peer-to-peer lending market. *Electronic Commerce Research and Applications*, 26:119–130.

Ma, X., Sha, J., Wang, D., Yu, Y., Yang, Q., and Niu, X. (2018). Study on a prediction of p2p network loan default based on the machine learning lightgbm and xgboost algorithms according to different high dimensional data cleaning. *Electronic Commerce Research and Applications*, 31:24–39.

Mammadli, S. (2016). Fuzzy Logic Based Loan Evaluation System. *Procedia Computer Science*, 102:495–499.

Martens, D., De Backer, M., Haesen, R., Vanthienen, J., Snoeck, M., and Baesens, B. (2007). Classification with ant colony optimization. *IEEE Transactions on Evolutionary Computation*, 11(5):651–665.

Millard, C. (2018). Blockchain and law: Incompatible codes? *Computer Law & Security Review*, 34(4):843–846.

Modi, R. (2018). Solidity Programming Essentials.

Moro, S., Cortez, P., and Rita, P. (2014). A data-driven approach to predict the success of bank telemarketing. *Decision Support Systems*, 62:22–31.

Network, E., Contracts, S., and Correspondent, M. (2018). ETHLend . io White Paper - Democratizing. (2).

Nizar, M. A. et al. (2017). Teknologi keuangan (fintech): Konsep dan implementasinya di indonesia [financial technology (fintech): It's concept and implementation

in indonesia]. Technical report, University Library of Munich, Germany.

Norta, A. and Leiding, B. (2019). Lowering Financial Inclusion Barriers With a Blockchain-Based Capital Transfer System. *Infocom*, pages 1–6.

Orús, R., Mugel, S., and Lizaso, E. (2019). Quantum computing for finance: Overview and prospects. *Reviews in Physics*, 4:100028.

Pokorná, M. and Sponer, M. (2016). Social Lending and Its Risks. *Procedia - Social and Behavioral Sciences*, 220:330–337.

Sicilia, M. A., García-Barriocanal, E., Sánchez-Alonso, S., and Cuadrado, J. J. (2019). Decentralized Persistent Identifiers: A basic model for immutable handlers. *Procedia Computer Science*, 146:123–130.

Statista (2015). Value of global peer to peer lending from 2012 to 2025.

Suliman, A., Husain, Z., Abououf, M., Alblooshi, M., and Salah, K. (2019). Monetization of iot data using smart contracts. *IET Networks*, 8(1):32–37.

Tang, H. (2019). Peer-to-peer lenders versus banks: substitutes or complements? *The Review of Financial Studies*, 32(5):1900–1938.

Taylor, P. J., Dargahi, T., Dehghantanha, A., Parizi, R. M., and Choo, K.-K. R. (2020). A systematic literature review of blockchain cyber security. *Digital Communications and Networks*, 6(2):147–156.

Thakre, A., Thabtah, F., Shahamiri, S. R., and Hammoud, S. (2020). A novel block chain technology publication model proposal. *Applied Computing and Informatics*.

Token, A. (2018). WeTrust Whitepaper Table of Contents. *Bravenewcoin.Com*, 1.

Tran, K. C. (2019). Ultimate Guide to Ethereum Lending: ETHLend, MakerDAO, BlockFi, SALT, Dharma & Compound.

Tunç, A. (2019). Feature Selection in Credibility Study for Finance Sector. *Procedia Computer Science*, 158:254–259.

Uriawan., W., Badr., Y., Hasan., O., and Brunie., L. (2022). Trustlend: Using borrower trustworthiness for lending on ethereum. In *Proceedings of the 19th International Conference on Security and Cryptography - SECRYPT,*, pages 519–524. INSTICC, SciTePress.

Uriawan., W., Hasan., O., Badr., Y., and Brunie., L. (2021). Collateral-free trustworthiness-based personal lending on a decentralized application (dapp). In *Proceedings of the 18th International Conference on Security and Cryptography - SECRYPT,*, pages 839–844. INSTICC, SciTePress.

Uriawan, W., Hasan, O., Badr, Y., and Brunie, L. (2022). Laps: Computing loan default risk from user activity, profile, and recommendations. In *2022 Fourth International Conference on Blockchain Computing and Applications (BCCA)*, pages 167–172. IEEE.

Uriawan, W., Wahana, A., Slamet, C., and Asih, V. S. (2021). A dapp architecture for personal lending on blockchain. In *2021 7th International Conference on Wireless and Telematics (ICWT)*, pages 1–6. IEEE.

Xia, Y., Yang, X., and Zhang, Y. (2018). A rejection inference technique based on contrastive pessimistic likelihood estimation for p2p lending. *Electronic Commerce Research and Applications*, 30:111–124.

Xie, Y., Holmes, J., and Dagher, G. G. (2020). Zerolender: trustless peer-to-peer bitcoin lending platform. In *Proceedings of the Tenth ACM Conference on Data and Application Security and Privacy*, pages 247–258.

Zhang, X., Aranguiz, M., Xu, D., Zhang, X., and Xu, X. (2018). Utilizing blockchain for better enforcement of green finance law and regulations. In *Transforming Climate Finance and Green Investment with Blockchains*, pages 289–301. Elsevier.

Zhang, Y., Wang, T., and Yuen, K.-V. (2022). Construction site information decentralized management using blockchain and smart contracts. *Computer-Aided Civil and Infrastructure Engineering*, 37(11):1450–1467.