Blockchain and Smart Contract

Term Project – Lending Service

107356003 廖信堯

Contents

◆ Introduction to customer lending service

Customer lending service, is identical to the lending service offered by banks, but lenders are not limited to financial institutes only. So far, large scale lending service are provided either by centralized banking institution or one-to-one manner. To get extra funds, customers have to either pay high fee or take worse loan condition and risk. However, relied on the new decentralized transactional model, blockchain technology provides a more cost-effective but reliable method to facilitate lending service. Decentralization fix some issues associated with current centralized system and employing smart contracts eliminate the hidden charges of middleman from traditional loan route (MinGi, 2018). By maintaining a distributed ledger through consensus algorithm, blockchain technology creates a trustworthy structure of lending service with transparent information and publicly accessibility.

◆ Motivation

In 2017–18, the trend of peer-to-peer lending service is popular or even out-of-control in China. Although government issued some regulations, now, there are still 2,677 active institutions. So far, the total loan insurance of P2P lending in China is over 8 trillion RMB (網貸之家, 2019). In United States, there is LendingClub which takes 45% P2P lending market share and has 47 billion loan insurance (LendingClub, 2019). Even in opaque environment with high fee and risk, customer lending is appealing and still growing. How about running the lending service on blockchain? No restriction of time, location and higher fees. Don’t have to deal with cumbersome procedures or auditing. As a result, I choose the topic to discuss the applicability of blockchain on customer lending service.

◆ Literature Review

Customer lending service or loan service is the process by which a company (mortgage bank, servicing firm, peer-to-peer etc.) collects interest, principal, and escrow payments from a borrower (Wikipedia, 2018). It includes large scale retail lending and peer-to-peer (P2P) lending. However, the global peer-to-peer lending market is expected to reach $1,000 billion by 2025, which attracts many fintech enthusiasts (SoftMediaLab, 2018). The P2P market is a field of unlimited financial opportunities. The combination of P2P market with blockchain technologies is an appealing idea for a startup or financial institution.

With the popularity of bitcoin, the awareness increased for the underlying technology of Bitcoin as well, which is blockchain (Moran, 2017). Blockchain was the technology that Bitcoin runs on, a distributed ledger system that keeps records of all transactions digitally (Hoffman, 2017). Businesses are adopting the blockchain technology and its distributed network to power their revolutionary ideas (Lemieux, 2016). On lending service, blockchain technology offers some advantages. One of them is no middleman (iOlite, 2018). Through the decentralized lending system, borrowers can reach out directly to lenders, eliminating the need for third-party intermediaries (banks, brokers). Another one is competitive rates and higher efficiency (iOlite, 2018). Replaced the bureaucracy of traditional system with blockchain, blockchain significantly cut operational costs and process time, which enable more competitive loan rates. Lower fees, better transparency, and saved time and money. Blockchain creates unbelievable conditions for lending procedure improvement

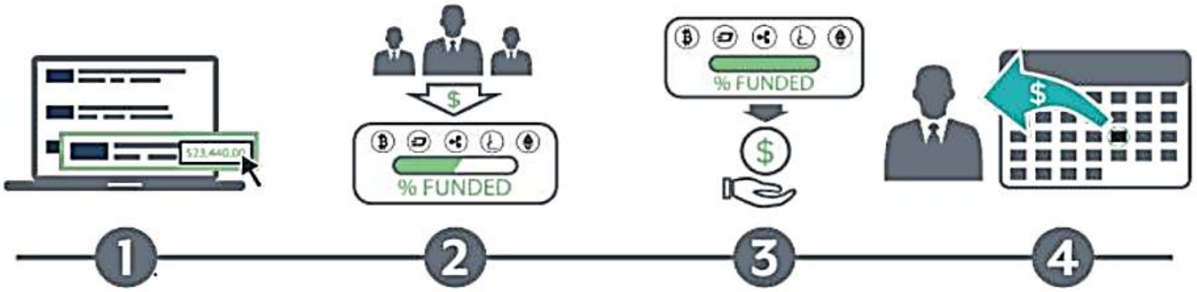
◆ Industry survey

Blockchain and smart contracts have taken the loans market like a storm, when they facilitate a peer-to-peer system built on trust, transparency and efficiency. It replaces the centralized lending model with the decentralized one. This new decentralized loans market is growing and is expected to reach as much as $897.85 billion by 2024 (compared with $26.16 billion in 2015) (Transparent Market Research, 2018). Considering the advantages for both sides, a number of blockchain lending projects are already being deployed.

Top lending projects on blockchain:

1. **SALT Lending** - Crypto-collateral lending

SALT is the largest blockchain lending platform, with a market cap of $126 million and roughly 55,500 holders. SALT is designed to enable its members to leverage their blockchain assets to secure cash loans, getting funds without having to sell their blockchain assets. There are no origination fees, closing costs, or prepayment penalties on loans arranged through the SALT platform. It chose a business model where fees charged to borrowers are rolled into an annual membership, which is an Ethereum-based ERC20 smart contract representing levels of access to the service.



To borrow money, members choose loans that posed by lenders and then put their collateral blockchain asset into a multi-signature smart contract to make credit agreement. No matter the value of the collateral depreciates or increase, borrowers will be notified to conduct some procedures, like revising loan term or loan balance, or the smart contract will initiate the liquidation of collateral automatically. As soon as the loan is repaid, the member's blockchain asset is returned.

2. **Ripio Credit Network (RCN)** - Global P2P cosigned lending

   RCN is a P2P global credit network protocol based on cosigned smart contracts and blockchain technology. It enables P2P lending using its RCN tokens. Unlike other P2P lending platforms, it has an alternative option instead of collateral, introducing an intermediary cosigner. RCN seeks to neutralize the lender's credit risk and, provide a mechanism for managing the debt collection. In exchange for a small fee, the lender assigns a portion of the credit risk to the cosigner. In the default scenario, the cosigner is able to use the fees collected from all the lenders to compensate them. RCN's primary focus is Latin America, where 63% of the population is unbanked. The company has a market cap of $45 million and approximately 6,600 holders.

3. **ETHend** - Global P2P lending

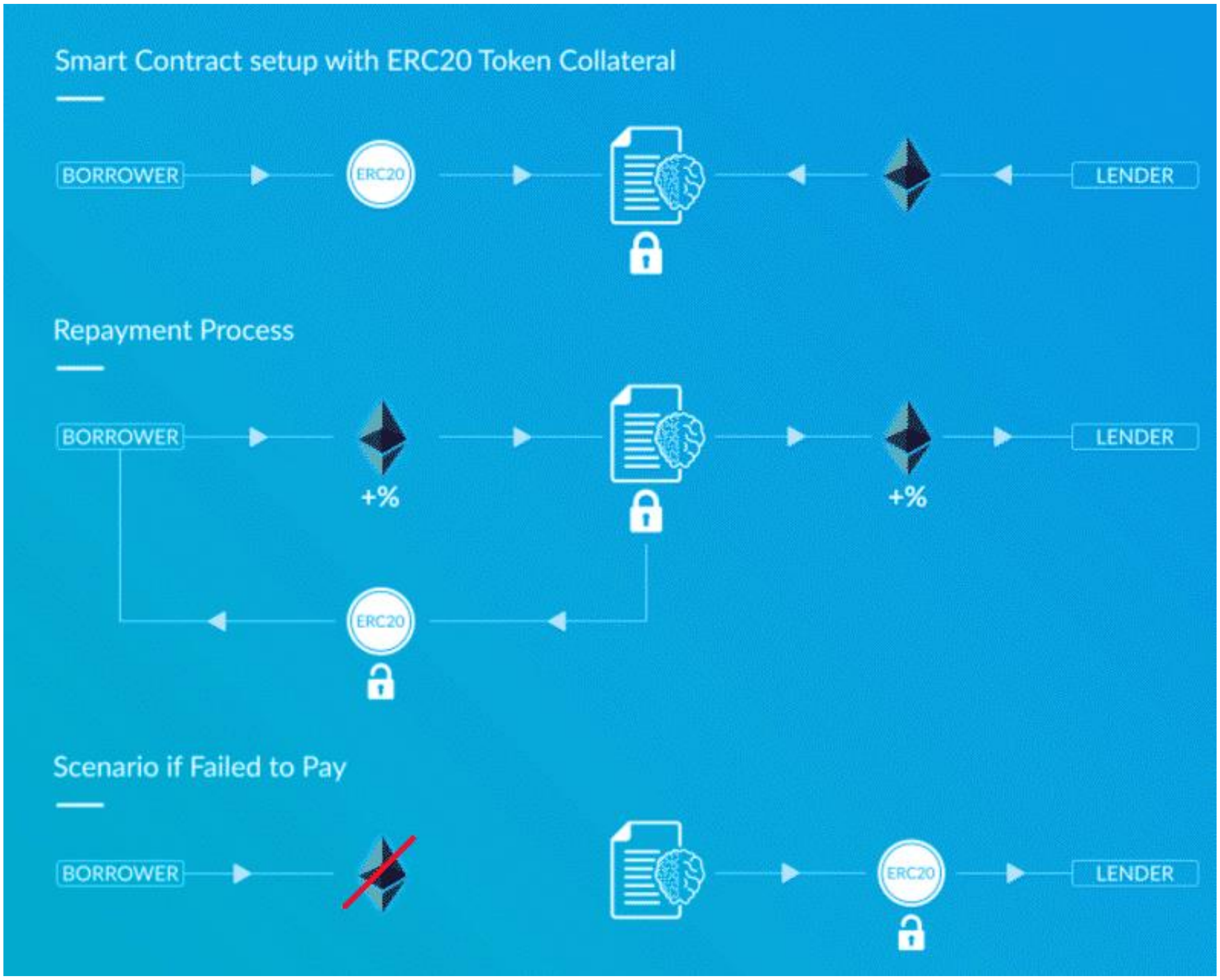
   ETHLend is an Ethereum-based decentralized lending platform. The loan terms, transfer of funds, and collateral are all handled automatically through smart contracts. ETHLend's market cap stands at $45 million, with roughly 172,000 holders. More details about the implementation of ETHLend will be discussed at next section.

- **Blockchain framework of ETHLend**

ETHLend initializes decentralized lending on Ethereum network by using ERC-20 compatible tokens as collateral. Since the pseudo-anonymous nature of Ethereum blockchain network, it led to the insufficient details of the borrower to enforce jurisdiction and increase the possibility to avoid repaying. With digital asset as collateral, ETHLend provides decentralized solutions to avoid loss of capital and to make a globally accessible lending market.

Compared with centralized borrowing system, decentralization changes the trust relationship between borrowers and loan providers. Loan collateral is locked and controlled by smart contracts that are broadcasted on the public blockchain, which creates a trustless environment. Besides, the public ledger is open for inspection. Every transaction is recorded and can be verified, which eliminates the blind trust with banking institutions. Not to mention the limitlessness on time and location, which increases the accessibility of service and create a much broader pool of loan liquidity.

So how does ETHLend work? Borrowers trigger a loan request with data like loan’s term, rate and collateral of tokens or Ethereum Name Service (ENS) domains. If a lender agrees to these conditions, a loan agreement will be created automatically by smart contract. After that, there are two scenarios. If the borrower repays the loan on time, the lender will receive the original amount plus interest. On the other hand, if the borrower fails to repay a loan, the lender will receive the collateral as compensation. As the bridge of borrower and lender, smart contracts perform a secured loan without relying on third parties. Moreover, since ERC-20 tokens can represent any value theoretically, smart contracts provide the possibility to control the value on blockchain. The whole process is displayed below (Sessa, 2018).



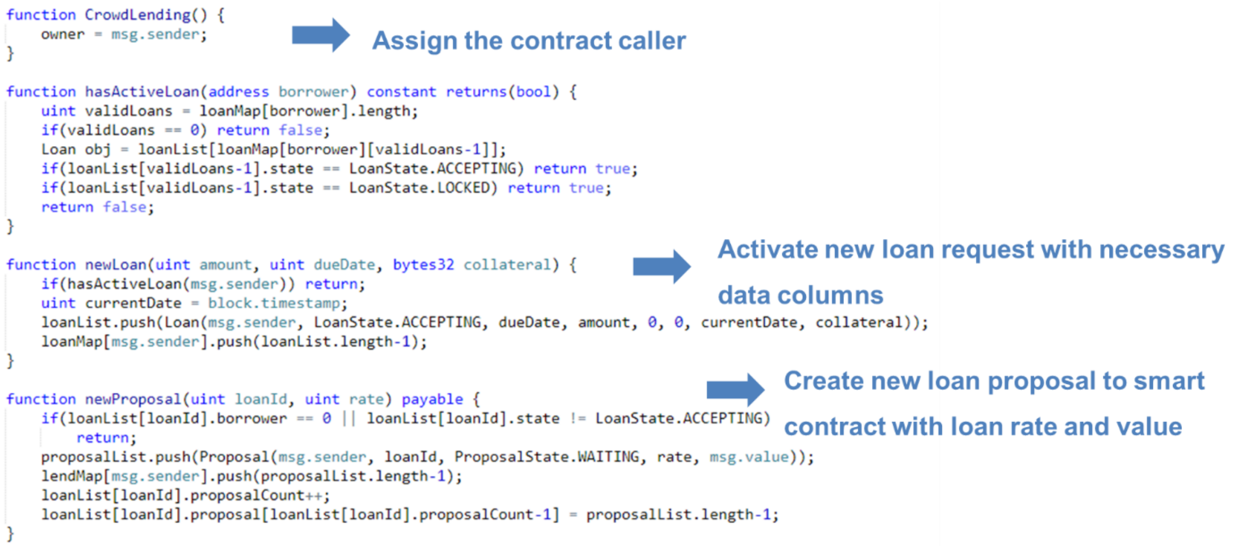
- **Discussion and suggested solution**

ETHLend provides a decentralized platform where secured, peer to peer Ethereum loans can take place between any two people in the world. Based on the framework of ETHLend, I think there are two points that are worthwhile to discuss further. First, the lending pool of secured loan is limited. People can borrow money from others only when they have ERC-20 tokens, which I think will largely restrict the growth of size of ETHLend lending pool. ETHLend knew the drawbacks of secured loan, so they proposed a solution called Credit tokens (CRE). Each time borrower repays a full debt amount, borrower will earn a CRE, which can represent the credit ability or reputation. Lenders can lend money to a borrower based on the CRE the borrower have, which makes unsecured loan feasible.

Secondly, ETHLend is a one-to-one lending platform. This property led to the situation that loan amount be restricted by the loaning capability of a lender. How about a many-to-many model? Like crowdfunding, the platform enables collecting funds from multiple lenders to increase loaning capability and distributing lender's risk of default to multiple bids. It seems that the many-to many business model are beneficial for both sides. As a result, based on the discussion mentioned above, I propose a many-to-many business model whose system structure is similar to ETHLend, but there are 3 differences:

1. The loan terms, including amount and interest rate, are determined by lenders instead of setting by borrowers.
2. Borrowers can accept multiple proposals within requested amount, which gives them rights to choose suitable conditions for them.
3. The collateral is not restricted to ERC-20 tokens only.

The following are the toy smart contract (link: https://github.com/JustinLiao93/BCSC) that designed by above concepts.

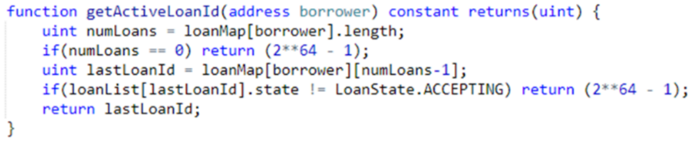


```
function CrowdLending() {
    owner = msg.sender;
}

function hasActiveLoan(address borrower) constant returns(bool) {
    uint validLoans = loanMap[borrower].length;
    if(validLoans == 0) return false;
    Loan obj = loanList[loanMap[borrower][validLoans-1]];
    if(loanList[validLoans-1].state == LoanState.ACCEPTING) return true;
    if(loanList[validLoans-1].state == LoanState.LOCKED) return true;
    return false;
}

function newLoan(uint amount, uint dueDate, bytes32 collateral) {
    if(hasActiveLoan(msg.sender)) return;
    uint currentDate = block.timestamp;
    loanList.push(Loan(msg.sender, LoanState.ACCEPTING, dueDate, amount, 0, 0, currentDate, collateral));
    loanMap[msg.sender].push(loanList.length-1);
}

function newProposal(uint loanId, uint rate) payable {
    if(loanList[loanId].borrower == 0 || loanList[loanId].state != LoanState.ACCEPTING)
        return;
    proposalList.push(Proposal(msg.sender, loanId, ProposalState.WAITING, rate, msg.value));
    lendMap[msg.sender].push(proposalList.length-1);
    loanList[loanId].proposalCount++;
    loanList[loanId].proposal[loanList[loanId].proposalCount-1] = proposalList.length-1;
}
```

Assign the contract caller

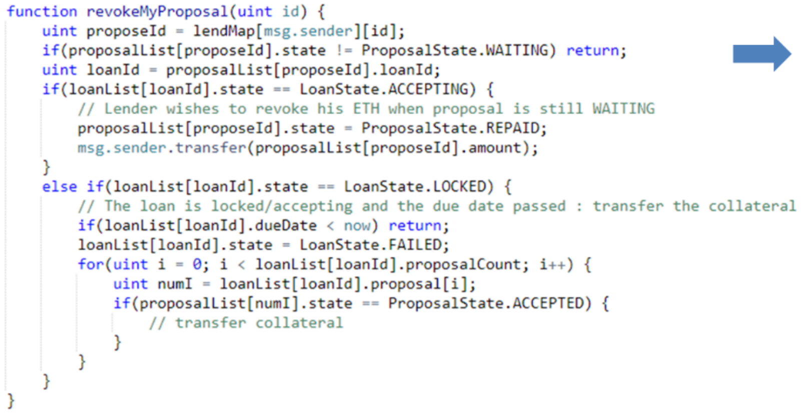
Activate new loan request with necessary data columns

Create new loan proposal to smart contract with loan rate and value



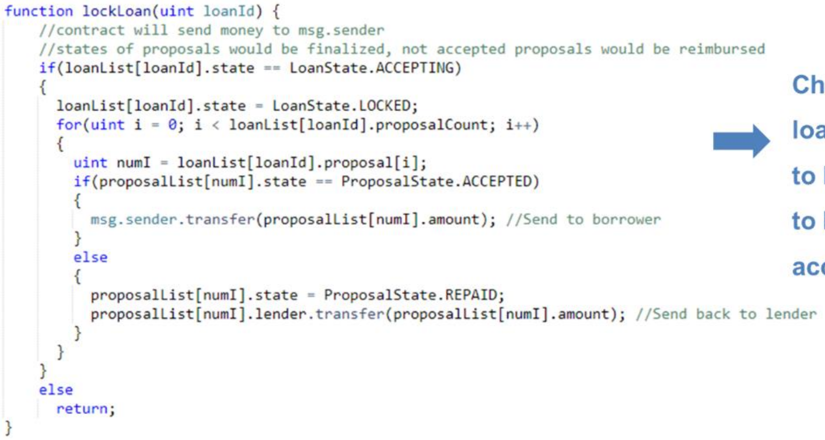
```
function getActiveLoanId(address borrower) constant returns(uint) {
    uint numLoans = loanMap[borrower].length;
    if(numLoans == 0) return (2**64 - 1);
    uint lastLoanId = loanMap[borrower][numLoans-1];
    if(loanList[lastLoanId].state != LoanState.ACCEPTING) return (2**64 - 1);
    return lastLoanId;
}
```

Check active LoanId and get the Id



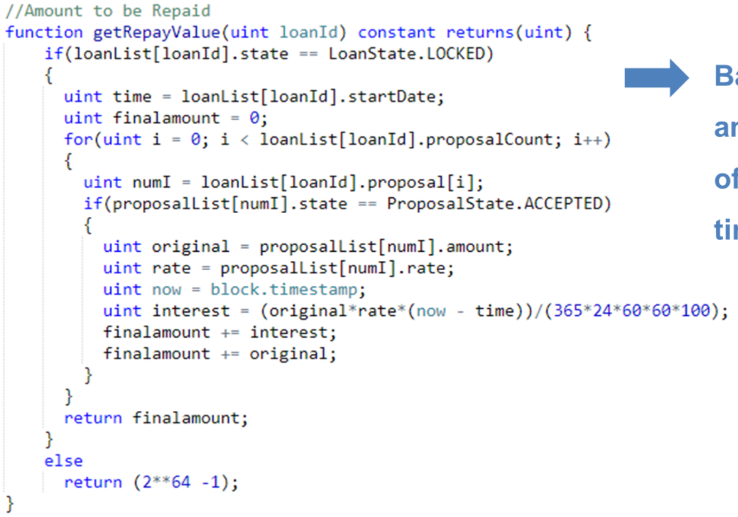
```
function revokeMyProposal(uint id) {
    uint proposeId = lendMap[msg.sender][id];
    if(proposalList[proposeId].state != ProposalState.WAITING) return;
    uint loanId = proposalList[proposeId].loanId;
    if(loanList[loanId].state == LoanState.ACCEPTING) {
        // Lender wishes to revoke his ETH when proposal is still WAITING
        proposalList[proposeId].state = ProposalState.REPAID;
        msg.sender.transfer(proposalList[proposeId].amount);
    }
    else if(loanList[loanId].state == LoanState.LOCKED) {
        // The loan is locked/accepting and the due date passed : transfer the collateral
        if(loanList[loanId].dueDate < now) return;
        loanList[loanId].state = LoanState.FAILED;
        for(uint i = 0; i < loanList[loanId].proposalCount; i++) {
            uint numI = loanList[loanId].proposal[i];
            if(proposalList[numI].state == ProposalState.ACCEPTED) {
                // transfer collateral
            }
        }
    }
}
```

Enable lenders to withdraw pending proposal or to request for collateral when the loan status is default



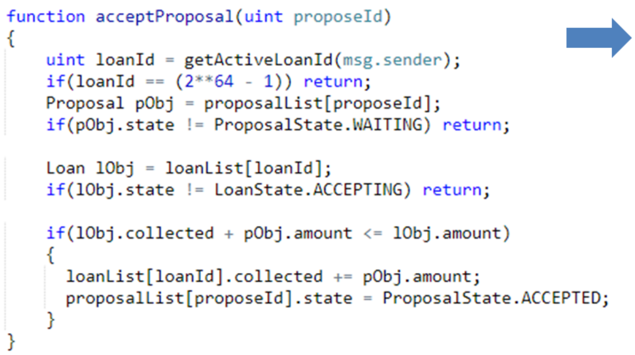
```
function lockLoan(uint loanId) {
    //contract will send money to msg.sender
    //states of proposals would be finalized, not accepted proposals would be reimbursed
    if(loanList[loanId].state == LoanState.ACCEPTING)
    {
      loanList[loanId].state = LoanState.LOCKED;
      for(uint i = 0; i < loanList[loanId].proposalCount; i++)
      {
        uint numI = loanList[loanId].proposal[i];
        if(proposalList[numI].state == ProposalState.ACCEPTED)
        {
          msg.sender.transfer(proposalList[numI].amount); //Send to borrower
        }
        else
        {
          proposalList[numI].state = ProposalState.REPAID;
          proposalList[numI].lender.transfer(proposalList[numI].amount); //Send back to lender
        }
      }
    }
    else
      return;
}
```

Check loan and proposal status; if a loan is accepting, change the status to locked and transfer the amount to borrower; if a proposal is not accepted, send ETH back to lenders



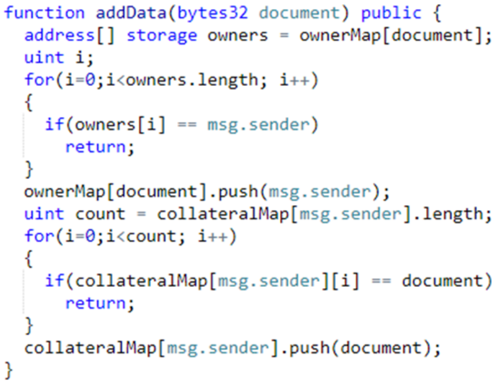
```
//Amount to be Repaid
function getRepayValue(uint loanId) constant returns(uint) {
    if(loanList[loanId].state == LoanState.LOCKED)
    {
      uint time = loanList[loanId].startDate;
      uint finalamount = 0;
      for(uint i = 0; i < loanList[loanId].proposalCount; i++)
      {
        uint numI = loanList[loanId].proposal[i];
        if(proposalList[numI].state == ProposalState.ACCEPTED)
        {
          uint original = proposalList[numI].amount;
          uint rate = proposalList[numI].rate;
          uint now = block.timestamp;
          uint interest = (original*rate*(now - time))/(365*24*60*60*100);
          finalamount += interest;
          finalamount += original;
        }
      }
      return finalamount;
    }
    else
      return (2**64 -1);
}
```

Based on the interest rate and amount, to calculate the amount of repay; use the block timestamp as the starting point



```
function acceptProposal(uint proposeId)
{
    uint loanId = getActiveLoanId(msg.sender);
    if(loanId == (2**64 - 1)) return;
    Proposal pObj = proposalList[proposeId];
    if(pObj.state != ProposalState.WAITING) return;

    Loan lObj = loanList[loanId];
    if(lObj.state != LoanState.ACCEPTING) return;

    if(lObj.collected + pObj.amount <= lObj.amount)
    {
      loanList[loanId].collected += pObj.amount;
      proposalList[proposeId].state = ProposalState.ACCEPTED;
    }
}
```

Based on status of loan and proposal, to finalize the loan request (proposal amount + collected amount must <= request loan amount)



```
function addData(bytes32 document) public {
  address[] storage owners = ownerMap[document];
  uint i;
  for(i=0;i<owners.length; i++)
  {
    if(owners[i] == msg.sender)
      return;
  }
  ownerMap[document].push(msg.sender);
  uint count = collateralMap[msg.sender].length;
  for(i=0;i<count; i++)
  {
    if(collateralMap[msg.sender][i] == document)
      return;
  }
  collateralMap[msg.sender].push(document);
}
```

Store the collateral info on the blockchain

- ## Conclusion

Blockchain builds the foundation for greater visibility into lending environments, and analytics tools enable transparency of lending. With smart contracts expected to become mainstream as early as the first half of the 2020s, the lending market has a lot of catching up to do. As a result, banks and credit unions all dive into blockchain and smart contracts to study the feasibility of implementing alternative lending methodologies. However, with all the merits including lower fees, better transparency, saved time and money, there is still not a killer lending applicability on the market and deploying lending environment on the blockchain is not prevailing at all. The reason behind the situation I guess it was caused by the unfamiliarity and distrust on blockchain technology. No matter how companies explain the advantages of their blockchain products, users are hard to understand, not to mention they can believe on it. Because the mechanism is revolutionized by the blockchain, the process and trust relationship are totally different from before, which could cause more difficulties to persuade them to accept. In conclusion, I totally agree that blockchain can improve the efficiency of lending environment, but the knowledge and trust gaps between technology and customers is a massive challenge for this applicability to be accepted and widely utilized now.

- **Reference**

MinGi Kweon. (2018). A Study of Blockchain Technology in Facilitating Lending Services with Distributed Risk Aversion. Review of Computational Science and Engineering, 4(1), 91-101.

LendingClub. (2019). LendingClub Statistics. Retrieved June 8, 2019, from LendingClub: https://www.lendingclub.com/info/statistics.action

網貸之家. (2019). 網貸之家平台成交數據. Retrieved June 9, 2019, from 網貸之家: https://shuju.wdzj.com/platdata-1.html

Wikipedia. (2018). Definition of Loan Servicing. Retrieved June 10, 2019, from Wikipedia: https://en.wikipedia.org/wiki/Loan_servicing

SocialMediaLab. (2018). Use of Blockchain for P2P Lending Platforms. Retrieved June 10, 2019, from SocialMediaLab: http://bit.ly/2F8BXds

Moran, J. (2017). Blockchain and Bitcoin Basics: A Beginners Guide to Blockchain, Bitcoin, Other Cryptocurrencies and Why It's Important to Learn Now! CreateSpace Independent Publishing.

Lemieux, V. L. (2016). Trusting records: is Blockchain technology the answer? Records Management Journal, 26(2), 110-139.

Hoffman, N. (2017). Blockchain: Everything You Need to Know about Blockchain Technology and How It Works. CreateSpace Independent Publishing.

iOlite. (2018). The Loan Markets and Opportunities in Blockchain. Retrieved June 10, 2019, from iOlite: https://iolite.io/static/pdf/iOlite-LoanPink.pdf

Sessa, J. (2018). What is ETHLend? | Beginner's Guide. Retrieved July 27, 2018, from Coin Central : https://coincentral.com/ethlend-beginner-guide/

ETHLend. (2018). White Paper - Democratizing Lending. Retrieved June 10, 2019, from ETHLend: http://bit.ly/2MCaIya

SALT. (2018). White Paper – SALT Blockchain-backed Loans. Retrieved June 10, 2019, from SALT: https://whitepaper.io/document/85/salt-whitepaper

RCN. (2019). RCN – How it works. Retrieved June 10, 2019, from RCN: https://ripiocredit.network/how-it-works

Coinswitch. (2018). What is Ripio Credit Network (RCN)? Retrieved June 10, 2019, from Coinswitch: http://bit.ly/2WGshx0

Transparent Market Research. (2016). Increasing Small Business Units to Act as Building Blocks for Peer-to-Peer Lending Market. Retrieved June 10, 2019, from Transparent Market Research: http://bit.ly/2XL5yRI

Ko Ohashi. (2018). Proof-of-concept example for a lending platform Part 3. Retrieved June 10, 2019, from Credit Hydra: http://bit.ly/2IMowB0