Malnati Josh , Affel Harrison W, Mizanur Raman

**Project name:** “LendMoney Financing“

**Team name:** LendMoney 2.0

**Team members:**

Harrison – Team Lead, Solidity developer, Front End Developer, Design

Josh – Testing, Design, Front End Developer

Rahman - Backend Developer, Solidity Developer, Design

**Team Github repo link:** https://github.com/Mizanur888/Blockchain.git

# **Project Description**

**What?**

LendMoney Financing is a crowdfunding platform based on the ethereum blockchain. It allows people to lend funds for an idea, charity, or start-up business. It is fully decentralized, secure, and can be used in a trustless fashion. This platform is unique as it does not rely on banks to lend funds, but insteads facilitates easy and secure lending between users in a peer-to-peer fashion. If users do not repay the funds to the lender, the amount plus an additional percentage will be deducted from the debted user. This percentage can change according to how much the lender trusts the debtor. Alternatively, the lender can forgo direct withdrawal from the debtor and instead place a recurring interest fee each month. If this fee raises above the total debtors balance, the full wallet balance will be paid to the lender.

**Why?**

Often times working with a bank on getting a loan can be a long and tedious process, halting the production of new inventions or companies. The process involves a large amount of legal and financial paperwork that is prone to human error and is not completely transparent. LendMoney aims to solve this issue by facilitating a sure-fire way for individuals to confidently loan money to others, negating the need for a traditional bank.

**How?**

LendMoney Financing is built on top of the Ethereum platform. Instances of the LendMoney smart contract will be deployed on Remix / Locally if possible. A web app will be built for the user to interact with the LendMoney smart contract. This will be using the web3.js interface along with either React or Angular. Users will be able to login with Google login, email, or a username and password set. This is possible through the use of aws Firebase NoSQL database. In detail, the frameworks and tools we expect to use are:

**Front End**

• HTML, LESS (or SASS)/CSS/SCSS

• JavaScript / typescript, Angular or React/React Native

**Back end**

• Web3.js

• Firebase

**Tools**

• Remix

• Visual Studio Code

• Git/GitHub

• Infura

• Trello

**Miscellaneous**

• Ganache CLI, MetaMask

# **Related work and key contributions**

<https://www.investopedia.com/tech/salt-secured-automated-lending-technology-blockchain/>

SALT (Secure Automated Lending Technology) is another blockchain technology focused on lending money but utilizes the blockchain assets as collateral, focusing on USD and other fiat currencies as the main financial lend.

Our key contributions are three folds:

1. Fully decentralized

2. It offers the lender the ability to fully adjust contract terms

3. It gives lenders the ability to change interest fees and collateral based on how much they trust (or don’t trust) the debtor.

4. It allows the debtor a wider options of loan providers

5. It is fully crypto, and does not rely on ‘real world’ currency, such as USD

# **Deliverables**

1. Web app

1. Login With Valid credentials
2. Give wallet information to LendMoney for sending and receiving money
3. After a successful loan, deposit the money into the debtors account
4. Smart contact that will hold the contract conditions, between the Lender and Receiver

2. Backend (Firebase)

Set of APIs to interact with LendMoney smart contract

1. Sign Up user Api
2. Login Api to check for valid user
3. Create account api ( where it will get necessary wallet info from user )
4. Crete transaction api ( to track the sending and receiving of money )
5. Finalize transaction api ( After checking pre-validation it will finalize transaction on the Blockchain)
6. Notify Api ( will notify user With the Smart contact Expiration Date )

3. Unit testing Jasmine, Karma for Angular or Jest for react

4. Documentation and testing

5. Proposal

6. Design

7. Presentation

8. Final report (Poster, white paper, …)

# **Plan**

Week 1: Write proposal, setup team Github repo., Produce design document: backend, smart contract, final decisions on tools to be used

Week 2: Produce design document: frontend, workflow

Week 3: Implement smart contract, frontend, and backend

Week 4: Implement and test the flow

Week 5: Present the work

Week 6: Wrap up and deliver the final report.