Aim: to present an approach for maintaining and reporting credit history records on the Blockchain. In this approach, there exists no third party agency, the lending and borrowing contracts are based on smart contracts. The customer directly interacts with the lender or the bank and has complete control over disclosure of their records, and credit history. This approach also provides strong guarantee to the lenders, where they can check the correctness and completeness of the credit data disclosed.

Create a JavaScript front end and web3.js framework, where all data is stored in the blockchain. Both user and lenders’ front end has to be designed. In the back end, before performing a transaction, a smart contract has to be created between user and lender (use ethereum).

Pre-requisites: Know at least one programming language, and understand Data Structures. Under how classes and objects work. Basics of web development using HTML and CSS.

Things to learn to complete the project:

1. Familiarize ourselves with all the term being used
   1. Blockchain: <https://blockgeeks.com/guides/what-is-blockchain-technology/>
   2. Smart Contract - <https://blockgeeks.com/guides/smart-contracts/>
   3. Ethereum - <https://blockgeeks.com/guides/ethereum/>
2. To have a better understanding on the project, go through the document “Bloom Protocol”. Link:- <https://hellobloom.io/whitepaper.pdf>
3. JavaScript – front end development is completely dependent of JavaScript code. There are many links from where you can learn, choose the one which suits you the best.
   1. W3schools - <https://www.w3schools.com/js/default.asp>
   2. Tutorial point - <https://www.tutorialspoint.com/javascript/>
   3. YouTube - <https://www.youtube.com/watch?v=PwsigsH4oXw>
4. Learn Ethereum to make Decentralized Apps. I plan to learn this by make an app by following a 5 parts tutorial, which would teach us about solidity, its implementation is smart contracts, web3.js and everything else required. Through this we will have a clear idea on dApps, from here on we can choose to gain in-depth knowledge on any topic we feel is necessary.
   1. Part 1: <https://hackernoon.com/ethereum-development-walkthrough-part-1-smart-contracts-b3979e6e573e>
   2. Part 2: <https://hackernoon.com/ethereum-development-walkthrough-part-2-truffle-ganache-geth-and-mist-8d6320e12269>
   3. Part 3: <https://hackernoon.com/ethereum-development-walkthrough-part-3-security-limitations-and-considerations-d482f05278b4>
   4. Part 4: <https://hackernoon.com/ethereum-development-walkthrough-part-4-tokens-and-ercs-68645cf2f73e>
   5. Part 5: <https://hackernoon.com/ethereum-development-walkthrough-part-5-making-a-dapp-4c2a3bbcd5e5>
5. After learning all these concepts, we can analyze the problem which we aim to solve and proceed by doing things that is required to be done.