**Project Proposal**

**Title: Design of Smart Contracts in Decentralized Systems**

**Background and Motivation**

Smart contracts help you exchange money, property, shares, or anything of value in a transparent, conflict-free way while avoiding the services of a middleman. In large companies, there is always a third-party company that manages the payroll of the employees so by using smart contracts we can circumvent the middleman. In the real world, these types of traditional agreements involve paperwork, but a report by the consulting firm Cap Gemini states that: “Classic financial contracts simply don’t work for the digital economy. Using a physical medium means delays and inefficiencies and increases the likelihood of mistakes and fraud.” So, by using smart contracts, it saves a lot of money paid as a fee to the third parties, increases security as the company’s employee database remains within the company and need not be shared with the third party, the delays and inefficiencies can be dealt.

**Research Questions and Objectives**

The following questions come up during the implementation of the project

1. Can the middleman be eliminated, or does he take up another role in the organization?
2. How hard is it to implement the verbal contract to smart contract and as the organization’s size changes constantly how hard is it to add or terminate a contract and push it into the blockchain network?
3. How profitable is to remove the middleman and hire a set of programmers that can perfectly execute the smart contracts as smart contracts is a new and ambiguous technology for everyone?

The objective is to build a payroll system using smart contracts so that the middleman is eliminated avoiding security risks, financial delays and to bring profits to the company.

**Methodology**

Smart contracts can be used with bitcoin, but the bitcoin is limited to currency. Ethereum replaces bitcoin's more restrictive language and replaces it with a language that allows developers to write their own programs accommodating a flexible architecture for smart contracts with more flexibility and an Ethereum transaction is confirmed in seconds compared to minutes in bitcoin.

To build smart contracts we build a DApp (Decentralized Application) using Solidity. Solidity is a contract-oriented, high-level language for implementing smart contracts. It is influenced by C++, Python, and JavaScript and is designed to target the Ethereum Virtual Machine (EVM). Solidity is statically typed, supports inheritance, libraries and complex user-defined types among other features.

DApp or a Decentralized Application has its backend code running on a decentralized peer-to-peer network. Contrast this with an app where the backend code is running on centralized servers. A DApp can have frontend code and user interfaces written in any language (just like an app) that can make calls to its backend. Solidity connects the DApp front-end application to the blockchain.

Remix provides an integrated development environment (IDE) for smart contract development. It focuses on the development and deployment of Solidity written smart contracts.

**Validation and Verification**

There are 3 components in this phase, the program code, natural language contract and shared understandings and intentions of the parties. Any smart contract involves these three components, each having a different role depending on the contracts. These questions help validate if the smart contract executes correctly.

1. Does the program correctly encode the written natural language contract?
2. Does the program do what it is intended to do?
3. Does the system operation as a whole perform without any error and give the correct output, if there are multiple programs?

# Bibliography

**[i]Solidity** [Online] / auth. Documentation Ethereum Solidity // Ethereum Solidity. - Ethereum,

2017. - 2018. - https://solidity.readthedocs.io/en/v0.4.21/.

**[ii]Trust in Smart Contracts is a Process, As Well** [Journal] / auth. Firas Al Khalil Tom

Butler, Leona O’Brien, and Marcello Ceci. - [s.l.] : University College Cork, 2017. - 1.

**[iii]Validation and Verification of Smart Contracts: A Research Agenda** [Journal] / auth.

Magazzeni Daniele. - [s.l.] : IEEE, 2017. - Vols. 50-57.

**[iv]What is a Decentralized Application?** [Online] / auth. Hertig Alyssa // Coindesk. -

Coindesk, January 1, 2018. - March 17, 2018. - https://www.coindesk.com/information/what-

is-a-decentralized-application-dapp/.