**Solulab Assignment Doc**

**Assignment Question:**

* **Create an ICO Smart Contract, with your custom ERC20 Token with the below details:**
  + **Total Supply of Token: 100 Million**
  + **Initial Value at $0.01 (Pre-sale Quantity: 30 Million)**
  + **2nd Sale Value at $0.02 (Seed Sale Quantity: 50 Million)**
  + **Final Sale for Remaining Tokens should be dynamically allocated.**

**Explanation -** Assignment consists of two smart contracts which are:

1. **CrowdSaleRushTokens.sol**
2. **RushTokens.sol**

NOTE: **RushTokens.sol** inherits **CrowdSaleRushTokens.sol.**

To make the smart contract functioning well, we would first have to deploy CrowdSaleRushTokens.sol smart contract. Once this smart contract is deployed we are able to Deploy the RushTokens.sol smart contract which is the core SC consisting of all the crowdsale and ICO functionalities.

**Below are the transaction hashes and contract addresses that have been mentioned -**

1. **CrowdSaleRushTokens.sol -**

**Contract Address -** 0xA504154e47910371a3aAd5D49C8b1E31A2f119cB

**Transaction Hash** 0xc4f52feb6f3e5875382f8e4d1c0bb961a5328fb497166553bd752c71d1c9e1df

1. **RushTokens.sol -**

Contract Address - 0x86fD301E5216aD9bcA2Ae915400BebE6a57355Ee

**Transaction Hash** 0x17426c3fdc5eecebb400abc476bf7ce5a86cb6816f8657747b1702781088e02e

These smart contracts have been deployed on **Rinkeby testnet.**

**NOTE:** To deploy smart contracts please use REMIX IDE. I haven’t used truffle or web3 & solc compiler to deploy the smart contracts because of time constraints.

**Constructor method argument values are -**

\_RATE - 500

\_WALLET - 0x148e772046B59f5A61ea0F0322110eCE5f6bb146

\_TOKEN - 0xD6aE8250b8348C94847280928c79fb3b63cA453e