

# EraSwap Smart Contracts

By NetObjex Inc

4th January 2019

## Preface

The smart contract for TimeAlly staking platform is completed which includes

- Eraswap Token Contract
- NRTManager Contract ( which distributes NRT to different actors)
- Staking Contract

## Softwares and General Info Prior testing.

- Firefox or Chrome with metamask
- Remix
- The contracts can be deployed to any testnet (rinkeby / ropsten)
- **Etherscan** can be used to see the corresponding transactions.
- You can pass in the deployed contract address into *ropsten.etherscan.io / rinkeby.etherscan.io* according to the testnet you deployed.
- You can also see the **NRTReleased** by watching each address which is fed into etherscan.

## Initial Setup

1. Deploy the eraswap token contract. It will transfer all **9100 million** tokens to the address which deployed contract.
2. Before deploying NRTManager , change the **ReleaseNRTtime** from ( 30 hours + 6 hours) to 5 minutes so that you can release NRT in 5 minutes which will correspond to NRT being released monthly for the convenience of testing. (Line no 723 and 677).
3. Deploy NRTManager contract and pass the eraswap token address and pass 8 addresses which corresponds to the following actors
  - a. NewTalentsAndPartnerships
  - b. PlatformMaintenance
  - c. MarketingAndRNR
  - d. KmPards

- e. ContingencyFunds
- f. ResearchAndDevelopment
- g. BuzzCafe
- h. TimeSwapper

**Example:**

```
[
  "0x5526B758117863bcf1cF558cE864CB99bdAC781c",
  "0xFFbB9F2b2fBE60374f41D65783687B5E347C5b34",
  "0xFF0991dD365A0959330659430D7fF653558e5B6F",
  "0x43D12FC54830b2704039bFE43f50A42dbcF8b8E8",
  "0x7041F7c401A9bAB295C6633697f2Af3DDEA5dA6b",
  "0x420e2Fb0f8Bc333a7D1C6a3E705984367386Fe0d",
  "0x8f28E76120e96CE72767b4eEBe7D9C445D37631A",
  "0xDC8553bb6dea3a2856de1D1008BB367e3ECC8538"
]
```

This will deploy the NRTManager contract and the account which is used to deploy will be the owner of the contract.

3. Deploy the staking contract and pass the eraswap token address and NRTManager contract address as constructor arguments. This will deploy the staking contract.
4. From the account which was used to deploy eraswap token, transfer 8281000000 EST to NRT Manager contract using metamask.
5. Pass the staking contract address to NRTManager by invoking setStakingContract in the NRTManager contract.

**Eraswap Token**

Play around with different functions in eraswap token , try out transferring burning tokens etc. 9100 million tokens are premixed and no more can be minted. Some of the functionalities include

- Transfer
- TransferFrom
- Approve
- IncreaseAllowance

- DecreaseAllowance
- Burn
- BurnFrom
- Pause
- addPauser etc and many more

## **NRTManager**

NRTManager is responsible for distributing the different tokens to different actors. In order to release NRT, you have to invoke **receiveMonthlyNRT** function to distribute the tokens. You can only invoke this function only once during every month (5 minutes for testing). You can see various NRT released in etherscan. As requested, we have included functionalities for changing different pool addresses, changing ownership and many more. You can play around those functions in remix and see the different working. Some of the functionalities includes

- releaseMonthlyNRT
- addSigner
- setBuzzCafe
- setContingencyFunds
- setKmpards
- setMarketingAndRNR
- etc and many more

## **Staking**

1. After the initial setup to start staking create a new account in metamask.
2. Add the eraswap token to the account and send some EST to the account.
3. Now to create a staking contract and allowance need to be initiated by the user (this is a standard procedure which has to be followed).
4. To create the allowance copy the staking contract address and open the eraswap token contract and inside select the **increaseAllowance**

option and inside spender field enter the staking contract address. Inside the addedValue field enter the amount to be staked and click **transact**.

5. After the allowance transaction is complete go to the staking contract and select **createStakingContract**. Inside the **amount** field enter the amount to be staked and inside the **isTwoYear** field enter true for 2 year staking contract, false for one year staking contract and click **transact**.
6. After the staking contract transaction is complete you will receive an OrderID which can be viewed in the transaction details.
7. This OrderID can be used for getting details about the contract by entering the OrderID in **StakingDetails**.
8. The details of the staking contracts can be viewed on EtherScan.

### **Winding up the staking contract**

1. For winding up the contract, select the metamask account from which the staking contract was deployed.
2. Inside the **windUpContract** enter the **OrderID** in the field and click **transact**.
3. Now the staking contract will end and the amount staked will be released equally in 104 weeks.
4. The details of the wound up contracts can be viewed on EtherScan.

### **Loan from Staking Contract**

1. For taking a loan from the staked amount select the metamask account from which the staking contract was deployed.
2. Inside the staking contract select the **takeLoan** option and enter the **OrderID** of the staking contract and click **transact**.
3. During the loan period NRT release for the staking contract will not happen.

## Repaying the loan

1. For repaying the loan select the metamask account from which the staking contract was deployed.
2. Inside the staking contract select the **rePayLoan** option and enter the **OrderID** of the staking contract and click **transact**.
3. This will initiate a transaction and the loaned amount with the interest will be repayed back with interest.