UDACITY

‹ Back to Blockchain Developer

Decentralized Star Notary

REVIEW

CODE REVIEW 7

HISTORY

▾ index.html 3

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0"
  <title>Star Notary</title>
  <link rel="stylesheet" type="text/css" href="style.css">

  <script src="https://cdn.jsdelivr.net/gh/ethereum/web3.js/dist/web3.m
</head>

<body>
<div class="container">
  <h1>Star Notary</h1>

  <div class="row">
    <label class="w-1/2 bold" for="star-name">Star name:</label>
    <input class="w-1/2" id="star-name" name="star-name" required>
  </div>

  <div class="row">
```

```
    <label class="w-1/2 bold" for="star-story">Star story:</label>
    <input class="w-1/2" id="star-story" name="star-story" required>
  </div>

  <div class="row">
    <label class="w-1/2 bold" for="star-ra">ra:</label>
    <input class="w-1/2" id="star-ra" name="star-ra" required>
  </div>

  <div class="row">
    <label class="w-1/2 bold" for="star-dec">dec:</label>
    <input class="w-1/2" id="star-dec" name="star-dec" required>
  </div>

  <div class="row">
    <label class="w-1/2 bold" for="star-mag">mag:</label>
    <input class="w-1/2" id="star-mag" name="star-mag" required>
  </div>

```

SUGGESTION

For better user experience, it would be better if you have a separate input field for the to token field can be confusing.

```
  <button id="claim-button" onclick="claimButtonClicked()">Claim Star</

  <h4>Get claimed star </h4>

  <div class="row">
    <label class="w-1/2 bold" for="star-tokenId">Token id:</label>
    <input class="w-1/2" id="star-tokenId" name="star-tokenId" requirec
  </div>

  <div class="row">
    <label id="star-token-result" class="w-full"></label>
  </div>

  <button id="info-button" onclick="infoButtonClicked()">Get star</butt
</div>

<script>
  if (typeof web3 !== 'undefined') {
    web3 = new Web3(web3.currentProvider) // what Metamask injected
  } else {
    // Instantiate and set Ganache as your provider
    web3 = new Web3(new Web3.providers.HttpProvider('http://localhost:8
  }
  // The default (top) wallet account from a list of test accounts
  web3.eth.defaultAccount = web3.eth.accounts[0]
  // The interface definition for your smart contract (the ABI)
```

```
var StarNotary = web3.eth.contract(
  [
    {
      "constant": true,
      "inputs": [
        {
          "name": "interfaceId",
          "type": "bytes4"
        }
      ],
      "name": "supportsInterface",
      "outputs": [
        {
          "name": "",
          "type": "bool"
        }
      ],
      "payable": false,
      "stateMutability": "view",
      "type": "function"
    },
    {
      "constant": true,
      "inputs": [
        {
          "name": "",
          "type": "uint256"
        }
      ],
      "name": "starsForSale",
      "outputs": [
        {
          "name": "",
          "type": "uint256"
        }
      ],
      "payable": false,
      "stateMutability": "view",
      "type": "function"
    },
    {
      "constant": true,
      "inputs": [
        {
          "name": "tokenId",
          "type": "uint256"
        }
      ],
      "name": "getApproved",
      "outputs": [
        {
          "name": "",
          "type": "address"
        }
      ],
      "payable": false,
      "stateMutability": "view",
      "type": "function"
    },
    {
      "constant": false,
      "inputs": [
        {
          "name": "to",
          "type": "address"
        },
        {
          "name": "tokenId",
          "type": "uint256"
        }
      ],
      "name": "approve",
      "outputs": [],
      "payable": false,
      "stateMutability": "nonpayable",
      "type": "function"
    },
    {
      "constant": true,
      "inputs": [
        {
          "name": "",
          "type": "uint256"
        }
      ],
      "name": "tokenIdToStarInfo",
      "outputs": [
        {
          "name": "name",
          "type": "string"
        },
        {
          "name": "ra",
          "type": "string"
        },
        {
          "name": "dec",
          "type": "string"
        },
        {
          "name": "mag",
          "type": "string"
        },
        {
          "name": "story",
          "type": "string"
        }
      ],
    "payable": false,
    "stateMutability": "view",
    "type": "function"
  },
  {
    "constant": false,
    "inputs": [
      {
        "name": "from",
        "type": "address"
      },
      {
        "name": "to",
        "type": "address"
      },
      {
        "name": "tokenId",
        "type": "uint256"
      }
    ],
    "name": "transferFrom",
    "outputs": [],
    "payable": false,
    "stateMutability": "nonpayable",
    "type": "function"
  },
  {
    "constant": false,
    "inputs": [
      {
        "name": "from",
        "type": "address"
      },
      {
        "name": "to",
        "type": "address"
      },
      {
        "name": "tokenId",
        "type": "uint256"
      }
    ],
    "name": "safeTransferFrom",
    "outputs": [],
    "payable": false,
    "stateMutability": "nonpayable",
    "type": "function"
  },
  {
    "constant": true,
    "inputs": [
      {
        "name": "tokenId",
        "type": "uint256"
        }
      ],
      "name": "ownerOf",
      "outputs": [
        {
          "name": "",
          "type": "address"
        }
      ],
      "payable": false,
      "stateMutability": "view",
      "type": "function"
    },
    {
      "constant": true,
      "inputs": [
        {
          "name": "owner",
          "type": "address"
        }
      ],
      "name": "balanceOf",
      "outputs": [
        {
          "name": "",
          "type": "uint256"
        }
      ],
      "payable": false,
      "stateMutability": "view",
      "type": "function"
    },
    {
      "constant": false,
      "inputs": [
        {
          "name": "to",
          "type": "address"
        },
        {
          "name": "approved",
          "type": "bool"
        }
      ],
      "name": "setApprovalForAll",
      "outputs": [],
      "payable": false,
      "stateMutability": "nonpayable",
      "type": "function"
    },
    {
      "constant": false,
      "inputs": [
        {
          "name": "from",
          "type": "address"
        },
        {
          "name": "to",
          "type": "address"
        },
        {
          "name": "tokenId",
          "type": "uint256"
        },
        {
          "name": "_data",
          "type": "bytes"
        }
      ],
      "name": "safeTransferFrom",
      "outputs": [],
      "payable": false,
      "stateMutability": "nonpayable",
      "type": "function"
    },
    {
      "constant": true,
      "inputs": [
        {
          "name": "owner",
          "type": "address"
        },
        {
          "name": "operator",
          "type": "address"
        }
      ],
      "name": "isApprovedForAll",
      "outputs": [
        {
          "name": "",
          "type": "bool"
        }
      ],
      "payable": false,
      "stateMutability": "view",
      "type": "function"
    },
    {
      "anonymous": false,
      "inputs": [
        {
          "indexed": true,
          "name": "from",
          "type": "address"
        },
        {
          "indexed": true,
          "name": "to",
          "type": "address"
        },
        {
          "indexed": true,
          "name": "tokenId",
          "type": "uint256"
        }
      ],
      "name": "Transfer",
      "type": "event"
    },
    {
      "anonymous": false,
      "inputs": [
        {
          "indexed": true,
          "name": "owner",
          "type": "address"
        },
        {
          "indexed": true,
          "name": "approved",
          "type": "address"
        },
        {
          "indexed": true,
          "name": "tokenId",
          "type": "uint256"
        }
      ],
      "name": "Approval",
      "type": "event"
    },
    {
      "anonymous": false,
      "inputs": [
        {
          "indexed": true,
          "name": "owner",
          "type": "address"
        },
        {
          "indexed": true,
          "name": "operator",
          "type": "address"
        },
        {
          "indexed": false,
          "name": "approved",
          "type": "bool"
        }
      ],
      "name": "ApprovalForAll",
      "type": "event"
    },
    {
      "constant": false,
      "inputs": [
        {
          "name": "_name",
          "type": "string"
        },
        {
          "name": "_ra",
          "type": "string"
        },
        {
          "name": "_dec",
          "type": "string"
        },
        {
          "name": "_mag",
          "type": "string"
        },
        {
          "name": "_story",
          "type": "string"
        },
        {
          "name": "_tokenId",
          "type": "uint256"
        }
      ],
      "name": "createStar",
      "outputs": [],
      "payable": false,
      "stateMutability": "nonpayable",
      "type": "function"
    },
    {
      "constant": false,
      "inputs": [
        {
          "name": "_tokenId",
          "type": "uint256"
        },
        {
          "name": "_price",
          "type": "uint256"
        }
      ],
      "name": "putStarUpForSale",
      "outputs": [],
      "payable": false,
      "stateMutability": "nonpayable",
      "type": "function"
    },
    {
      "constant": false,
      "inputs": [
        {
          "name": "_tokenId",
          "type": "uint256"
        }
      ],
      "name": "buyStar",
      "outputs": [],
      "payable": true,
      "stateMutability": "payable",
      "type": "function"
    },
    {
      "constant": true,
      "inputs": [
        {
          "name": "_ra",
          "type": "string"
        },
        {
          "name": "_dec",
          "type": "string"
        },
        {
          "name": "_mag",
          "type": "string"
        }
      ],
      "name": "checkIfStarExist",
      "outputs": [
        {
          "name": "",
          "type": "bool"
        }
      ],
      "payable": false,
      "stateMutability": "view",
      "type": "function"
    }
  ]
)
// Grab the contract at specified deployed address with the interface

// Rinkeby
//var starNotary = StarNotary.at('0xe757C17BDC5A0decC59eb31F3e1Fa29C2

// ganache
var starNotary = StarNotary.at('0xca10a17423c486b3f4f87b064cf622b9242

function claimButtonClicked () {
  web3.eth.getAccounts(function(error, accounts) {
    if (error) {
      alert(error)
      return
    }

    var account = accounts[0]

    var name = document.getElementById('star-name').value
    var ra = document.getElementById('star-ra').value
    var dec = document.getElementById('star-dec').value
    var mag = document.getElementById('star-mag').value
    var story = document.getElementById('star-story').value

    var tokenId = document.getElementById("star-tokenId").value;
```

AWESOME

Good! Now we have the tokenId as additional input!

```

    starNotary.createStar.sendTransaction(name, ra, dec, mag, story,
      from: account
    }, function(error, result) {
      if (!error) {
        alert(result)
        var starClaimedEvent = starNotary.Transfer()
        starClaimedEvent.watch(function(error, result) {
          if (!error) {
            alert('star claimed')
          } else {
            alert('watching if error on star claim')
          }
        })
      } else {
        alert(error)
      }
    })
  })
}

function infoButtonClicked () {
  web3.eth.getAccounts(function(error) {
    if (error) {
      alert(error)
      return
    }

    var tokenId = document.getElementById("star-tokenId").value;

    starNotary.tokenIdToStarInfo.call(tokenId, function(error, result
```

AWESOME

Works as expected!

```
    if (!error) {
     alert(result)
     document.getElementById('result').innerHTML = result
    } else {
     alert(error)
    }
   })
  })
 }
</script>
</body>
</html>
```

▸ smart_contracts/test/StarNotaryTest.js 3

▸ README.md 1

▸ style.css

▸ smart_contracts/truffle-config.js

▸ smart_contracts/migrations/2_deploy_contracts.js

▸ smart_contracts/migrations/1_initial_migration.js

▸ notes.txt

RETURN TO PATH

Rate this review