Building Internet Of TRust on Web3 with Googlenium.com

bjective

As the role of electronically published information is taking center stage in the decision-making process, it becomes increasingly important to provide a technology capable of verifying and confirming news and stories published by media sources on the web.

The following line of reasoning is assumed:

- A true story is a story based on facts
- Facts are confirmed events
- Events are actions that have happened in certain places at certain times
- Actual events are events confirmed by trusted witnesses.

The said technology will be able to provide an immutable record of event confirmed by a number of witnesses with dynamically evaluated level of trustfulness.

dea

In this perspective, one of the most distinctive blockchain abilities which is to deliver unaltered and always available records, could be utilized to evaluate trustworthiness of electronically published stories by employing smart contracts designed to confirm facts and events of the story.

isposition

To lay out an idea let's assume following:

- every real story is based on real events.
- every real event must be observed and uniquely identified by real witnesses.
- every real witness (person or entity) can be uniquely identified.
- event and witness may register a unique smart contract on the blockchain.
- witnesses can be rewarded for confirming a given event.
- claim of witnessing an event can be programmatically evaluated.
- algorithms to calculate event's "reality" level and witness "credibility" are available.
- based on events credibility scores, it is possible to evaluate how "true" the whole story is.

To be evaluated, an event will be recorded on the blockchain as a smart contract containing information about time, place, subject and short description (full description may be available outside of the blockchain and from different sources).

In addition to event identifications, an Event Smart Contract (ESC) also contains witness credentials associated with the event.

TrustWOrthiness Coefficient (TWOC) can be calculated as a median of all witnesses' TWOC.

By confirming an event and providing PRoof of Event (PREV), witness may be awarded by raising its own TWOC as well as gaining some blockchain tokens.

The tokens supplier could be a storyteller who is willing to include a reference to the Event SC into the story.

PREV can be provided by a trusted authority that is able to confirm that the ID registered with witness SC has a proven relationship to the event.

The story itself may gain the blockchain tokens donations from readers and referrers. The acquired tokens can be used not only for interacting with ESC but also for various other services i.e., hosting, advanced design features etc.

Implementation

About Googlenium.com

Googlenium.com is an online service dedicated to creating and managing personal web presence sites - places where people can record, organize, and keep memorable events and stories of their lives. They may choose to keep them private or share with selected family members and friends as well as make them available to the rest of the world.

The already available Googlenium events management mechanism has been adopted for utilizing Ethereum blockchain and smart contracts. The later will be used to register events and create incentives for registered members to gain and use them.

To promote the idea of registering events on blockchain, the concepts of "event ownership" and "event stake" are being suggested.

Specially minted EON tokens will be used in Events Smart Contracts to raise event stakes and bid for the event ownership.

We are assuming that there is an established relationship between an event and a person registering it and that the value that person is setting as a stake is serving as a proof of event.

Anyone with the sufficient amount of Ether can register an ESC. After submitting event contract to the blockchain, the event registrar may return to the event and set a stake in any amount os EON tokens as a proof of event.

The event validation mechanism and TWOC calculation are not a part of this example and may be discussed later.

There might be other people who have witnessed or have direct knowledge of the event and who would like to participate in raising the event stake. They may do this in two ways - by sending tokens in amount less than current event stake and become witnesses or they may overbid current event owner by sending more than is currently at stake and become event owners.

The incentive for all participants is that difference between current stake and new ownership stake will be divided between previous owner and witnesses.

This is not the only incentive that can be used in ESC - i.e. authors who are referring to the event in their publications may be willing to prove them by referencing correspondent ESC and paying some tokens that also may be divided between ESC participants and raising event's stake.

Building a sample case

We will use Ethereum single node private net blockchain and Nethereum integration libraries to build interface between Events created by Googlenium users and Event Smart Contracts (ESC) registered on this blockchain implementation.

ESC will operate with the following notions and values:

S_i - *Initial Stake* = Initial stake set by the Event Registrar - person who defined event on Googlenium and registered ESC on the blockchain.
Registrar's (first witness W_1) Ethereum address and Event description URL as well as email address are saved within the ESC.

S_o - *Ownership Stake* = $S_i + S\Delta$

Δ - *Stake differential (delta)* >= 1 EON

S_{Wn} - *Witness Stake* = Witness n balance.

S_{esc} - *Event Stake* = ESC balance = $\sum_{i=1}^{n} S_{wi}$

B - Bid amount - amount of crypto being added to the witness stake $S_{wi.}$

There are two possible scenarios of how the bid amount can be distributed:

1). If witness new balance is more than S_o ($S_{wi} + B > S_o$), witness becomes new Event owner and Ownership Stake becomes Initial Stake $S_n = S_o$.

Δ — difference between ownership stake S_o and initial stake S_i is being distributed between event witnesses in following proportions:

- Previous owner receives 50% of the sum added to the previous ownership stake **S_{wn-1} += Δ / 2**
- New owner receives 25% of Δ amount **S_o += B +** $\Delta/4$
- The remaining quarter of Δ is distributed between all witnesses proportionally to their stakes

  Swi += ((100 * Swi)/ Sesc)* (Δ / 4) / 100

2). Witness' new balance is less or the same as ownership stake **$S_{wi} + A_b <= S_o$**

In this case witness' stake will be increased by the bid amount. ESC balance will be **$S_{wi} += A_b$**

The benefit for the witness would be that next time ownership is transferred, reward will be calculated proportionally to the new witness stake.

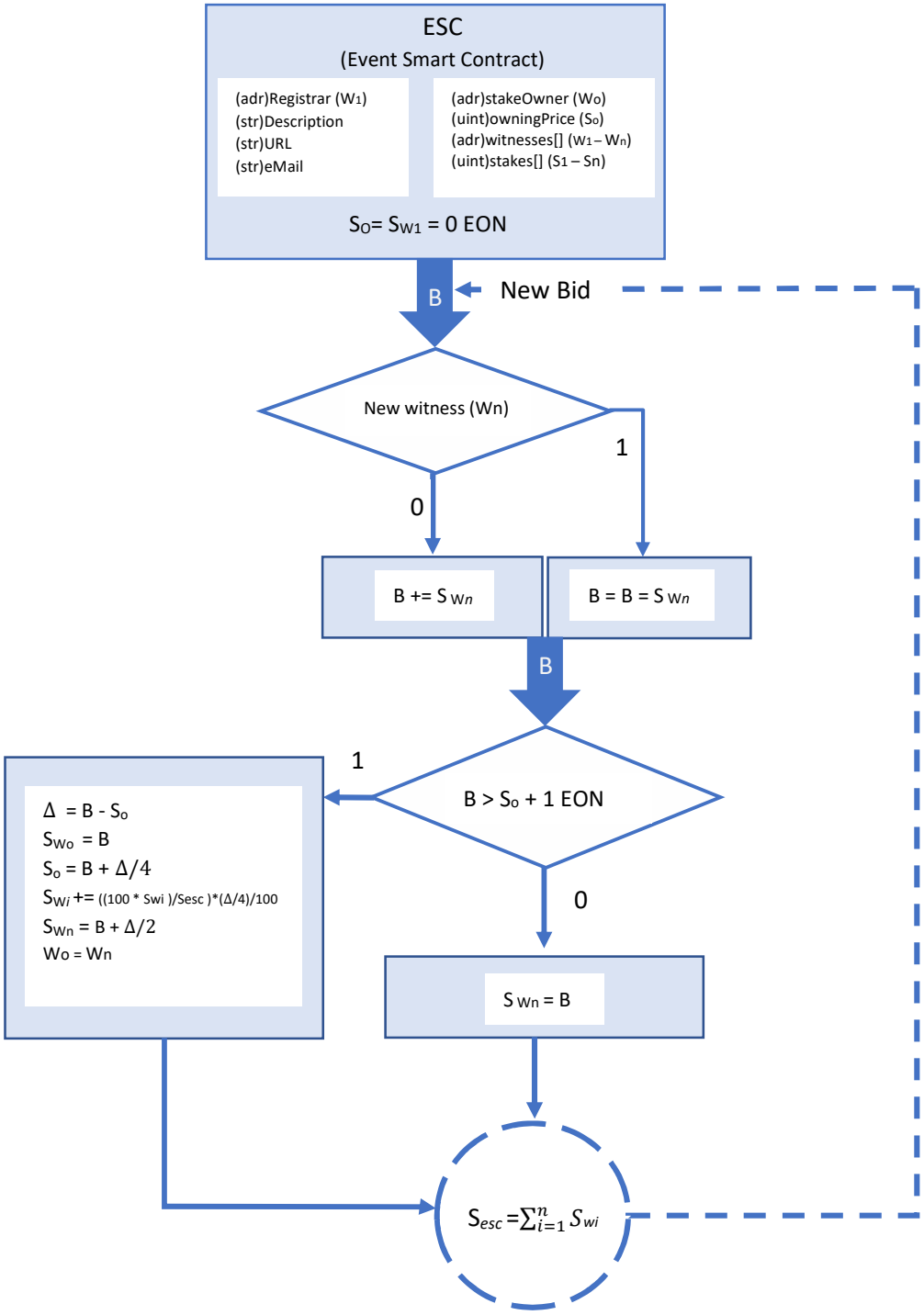
Some used and suggested terminology:

SC - Smart Contract

ESC - Event SC registered on blockchain representing combination of time, place, subject and description identifying unique witnessed occurrence

PREV - Proof of Event

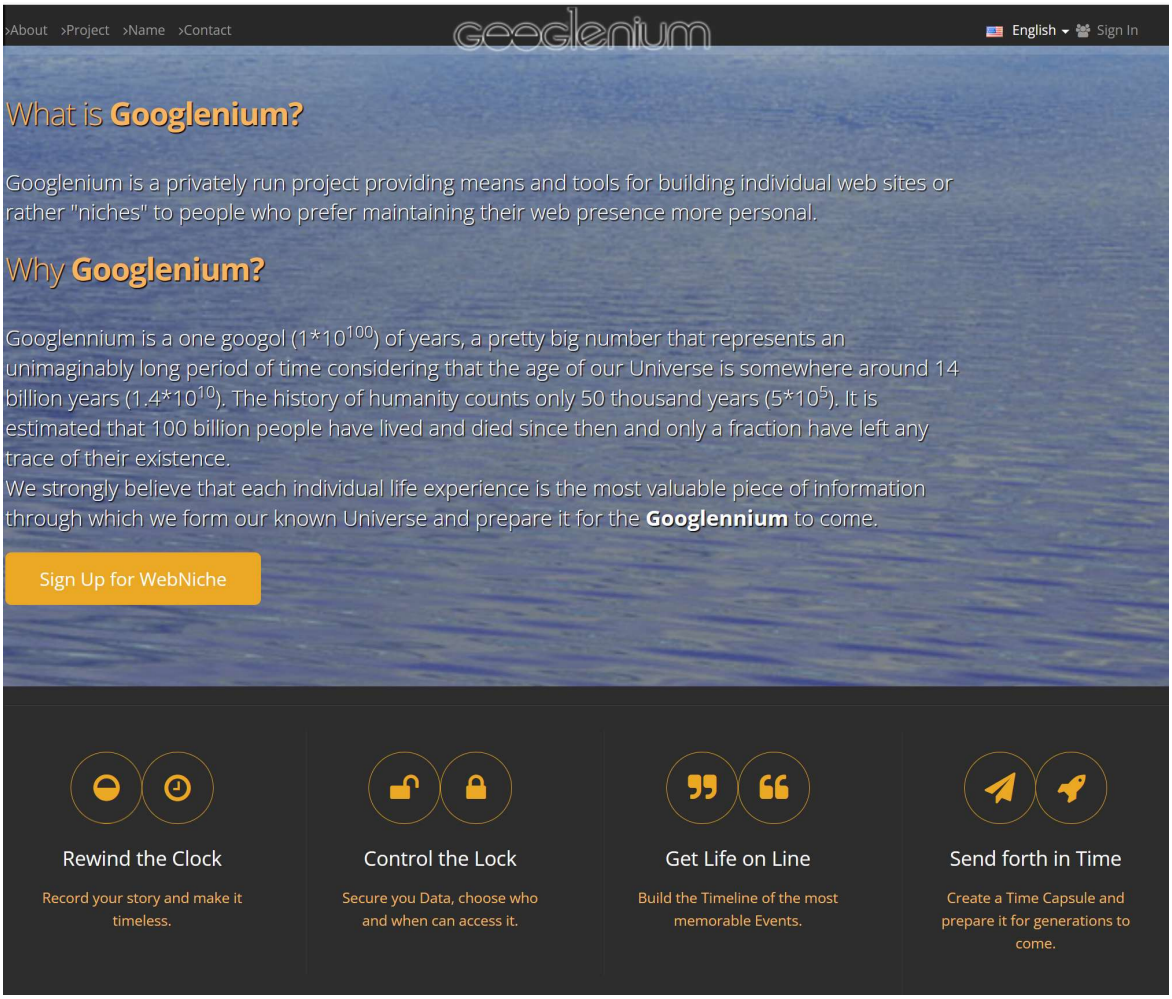
TWOC - TrustWOrthy Coefficient. A calculated number based on the number of confirmed witnessed events or objects.


ESC
(Event Smart Contract)
(adr)Registrar (W1)
(str)Description
(str)URL
(str)eMail
(adr)stakeOwner (Wo)
(uint)owningPrice (So)
(adr)witnesses[] (W1 – Wn)
(uint)stakes[] (S1 – Sn)
SO= SW1 = 0 EON
B
New Bid
New witness (Wn)
1
0
B += S Wn
B = B = S Wn
B
1
B > So + 1 EON
0
Δ = B - So
SWo = B
So = B + Δ/4
SWi += ((100 * Swi)/Sesc)*(Δ/4)/100
SWn = B + Δ/2
Wo = Wn
S Wn = B
Sesc =∑i=1n Swi

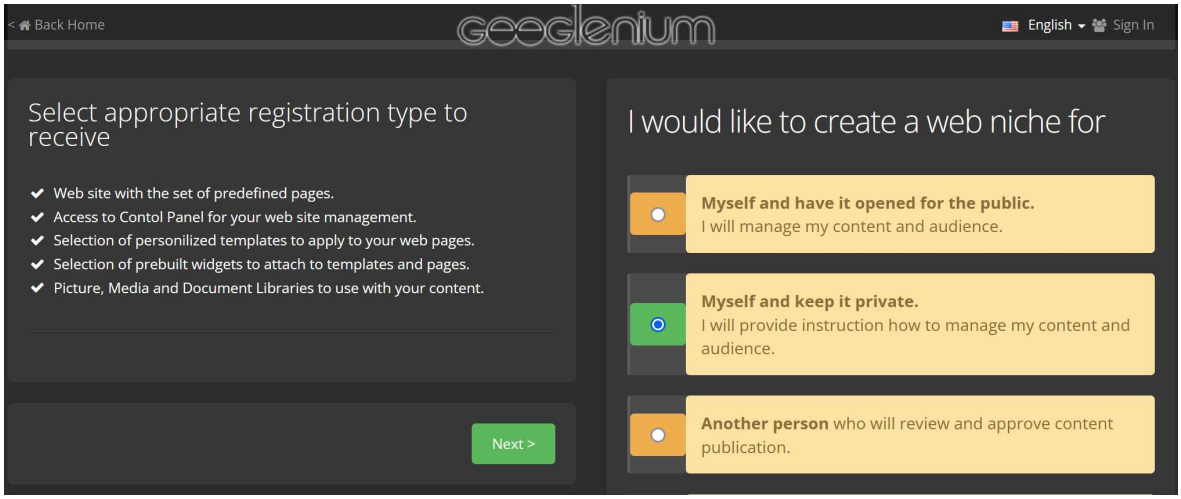
Example case

Let's start with registering a new user and creating a Googlenium 'niche' where we can later publish a story and deploy a smart contract.

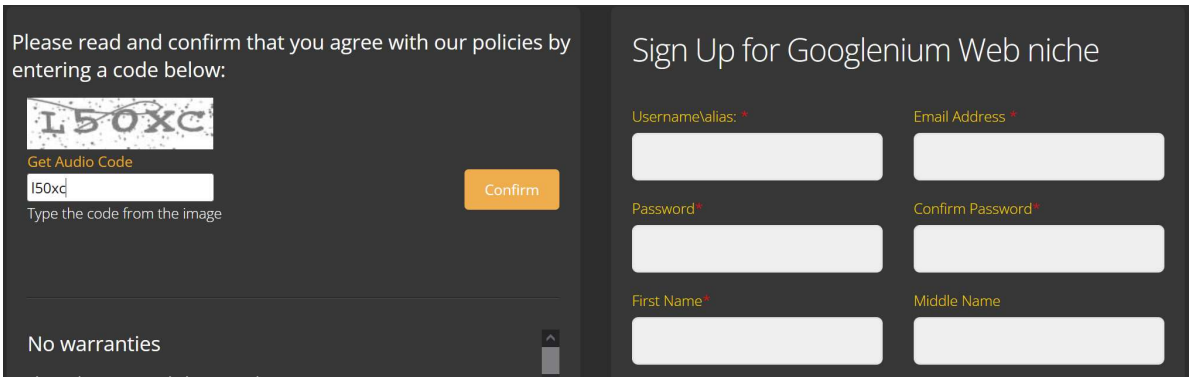
Navigate to https://googlenium.com site and scroll down to "Sign Up for WebNiche":



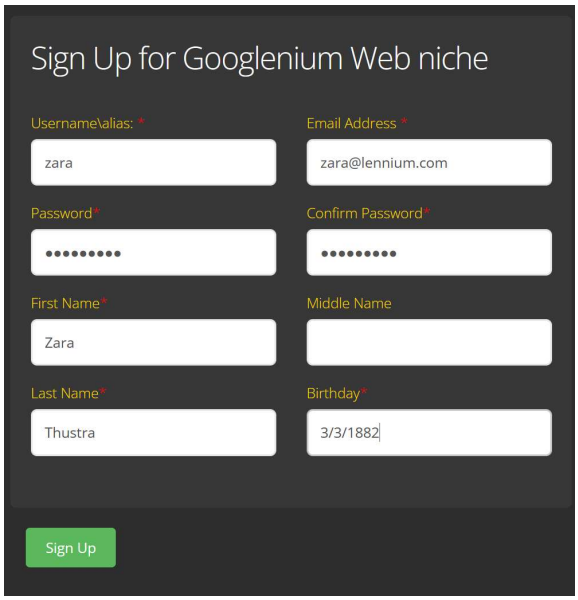
Click the button and select the type of site you will be creating:



Type the captcha code

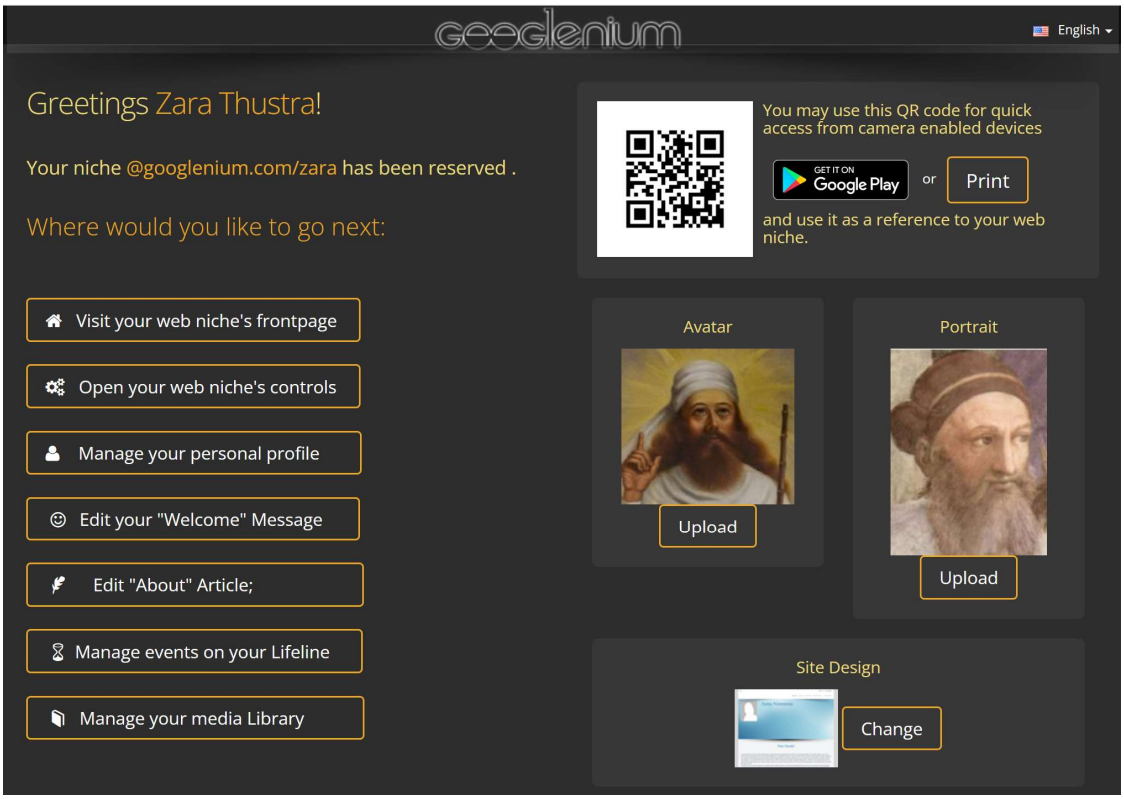


Enter new user data and sign up:

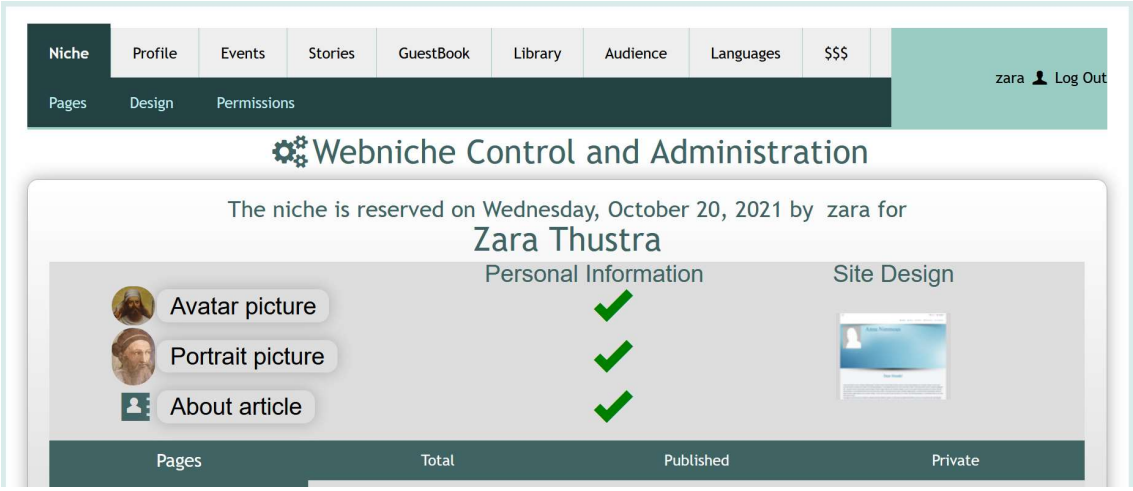


Wait a bit:

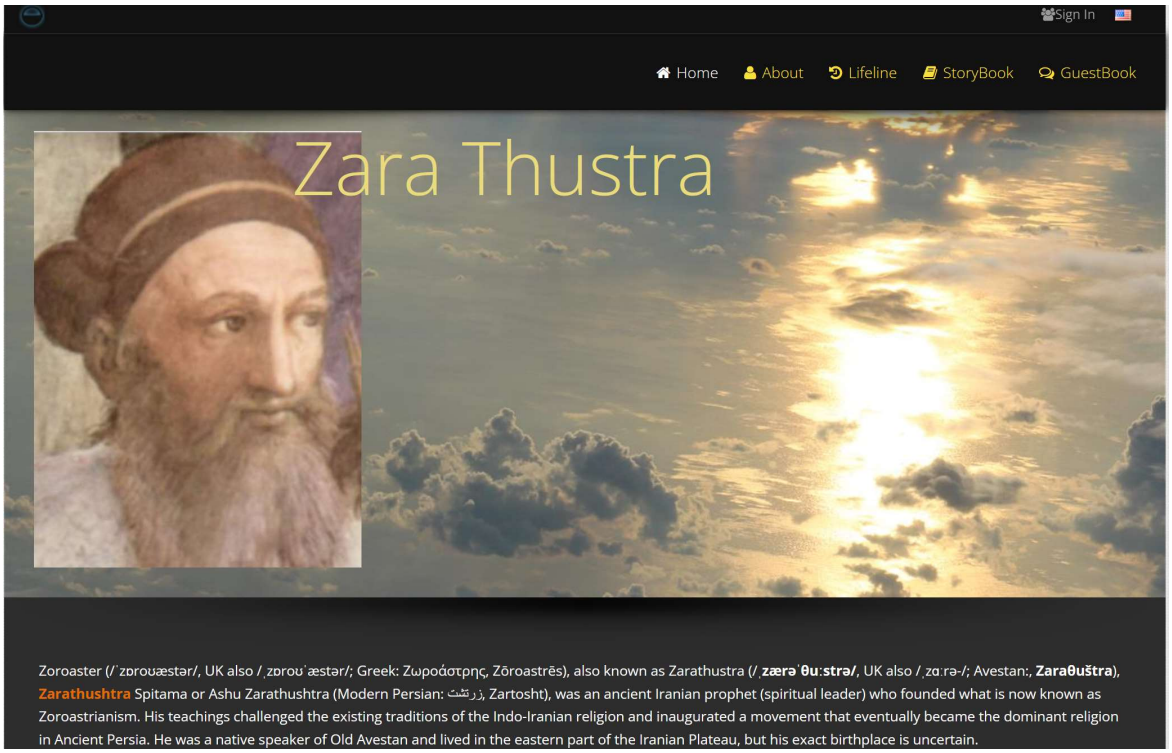
and new site will be ready.



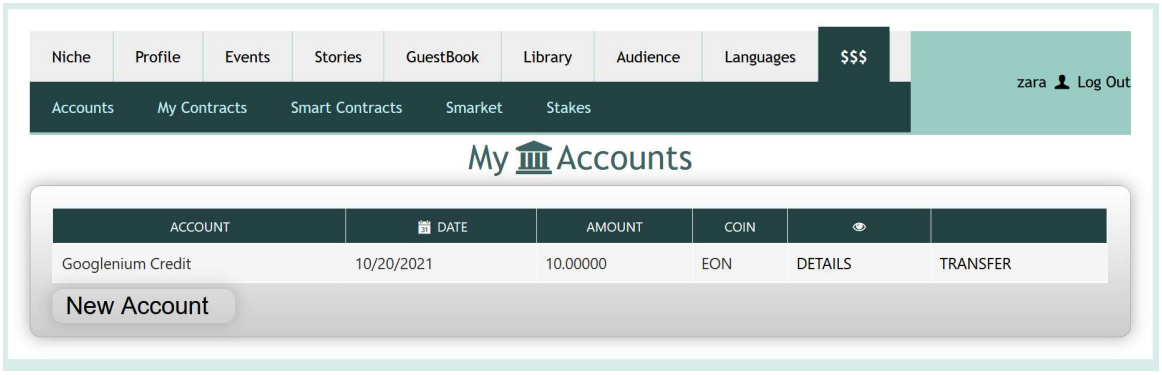
You may use available controls to upload pictures, manage site design and add events to your timeline.

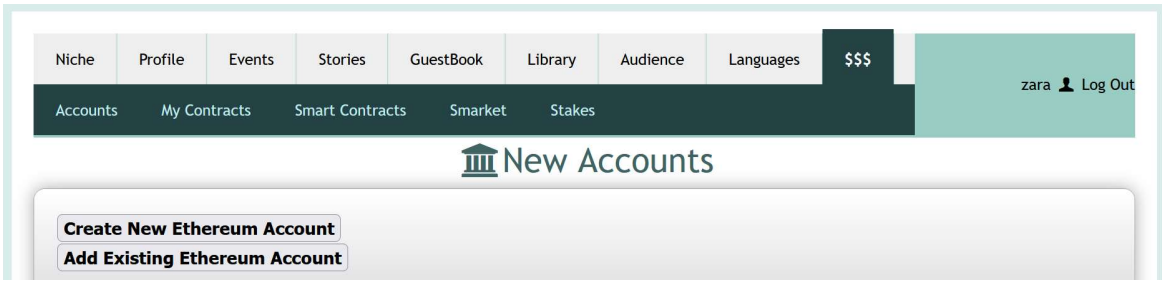


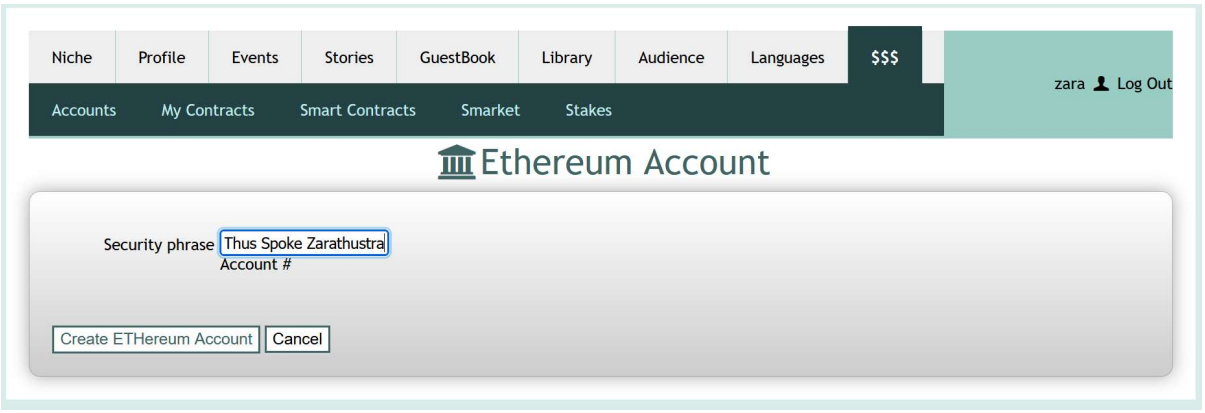
The niche and site are now ready for adding new stories and events.

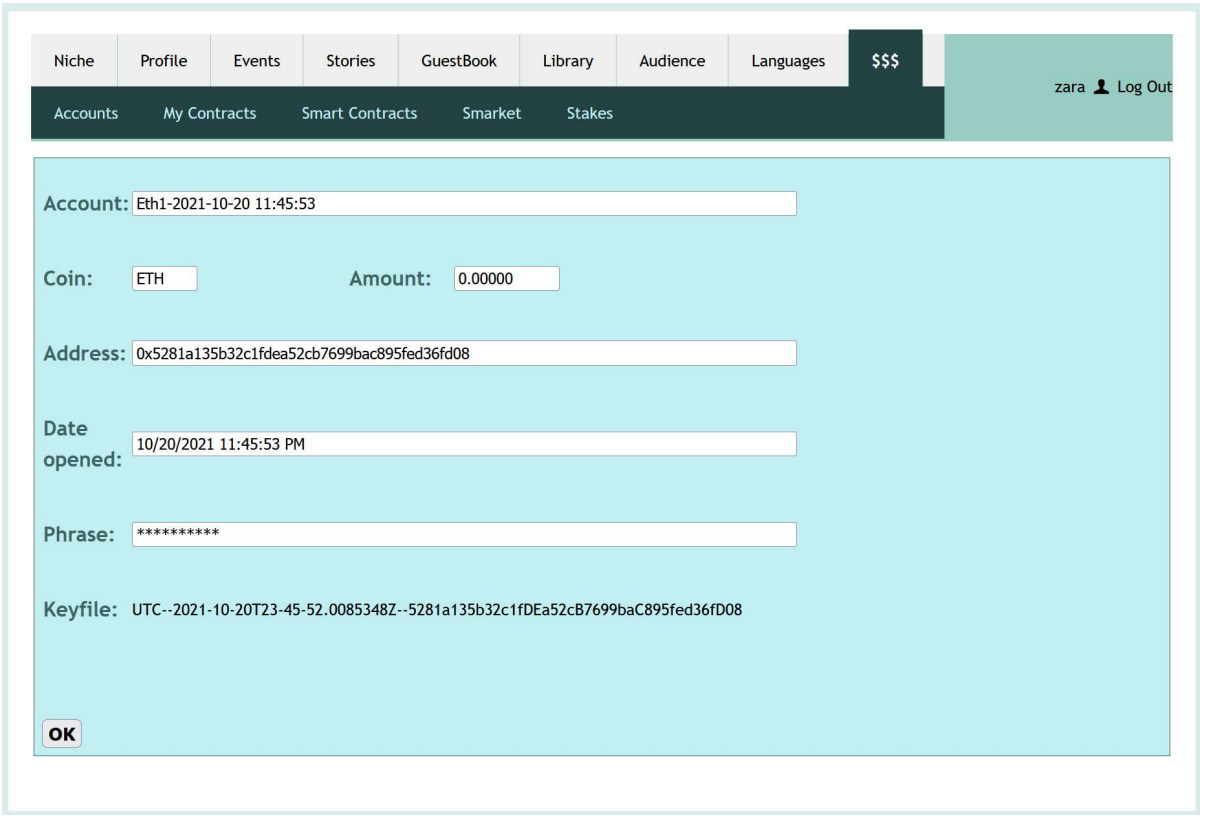


but before adding new Event and registering it on blockchain, we will create blockchain account:



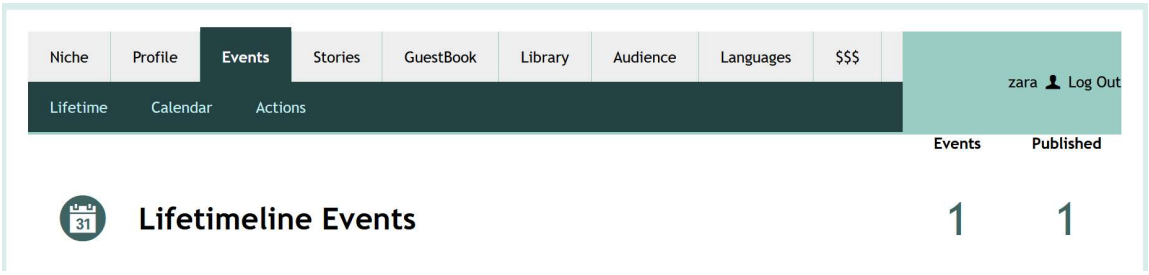


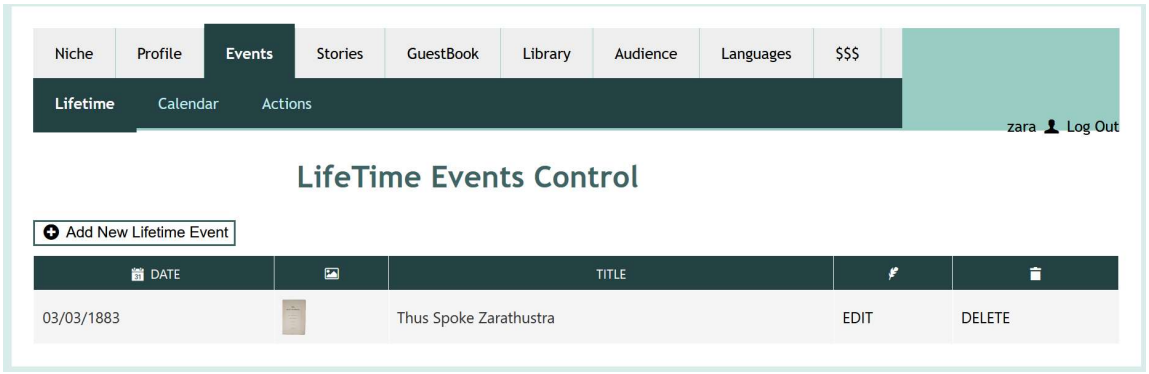


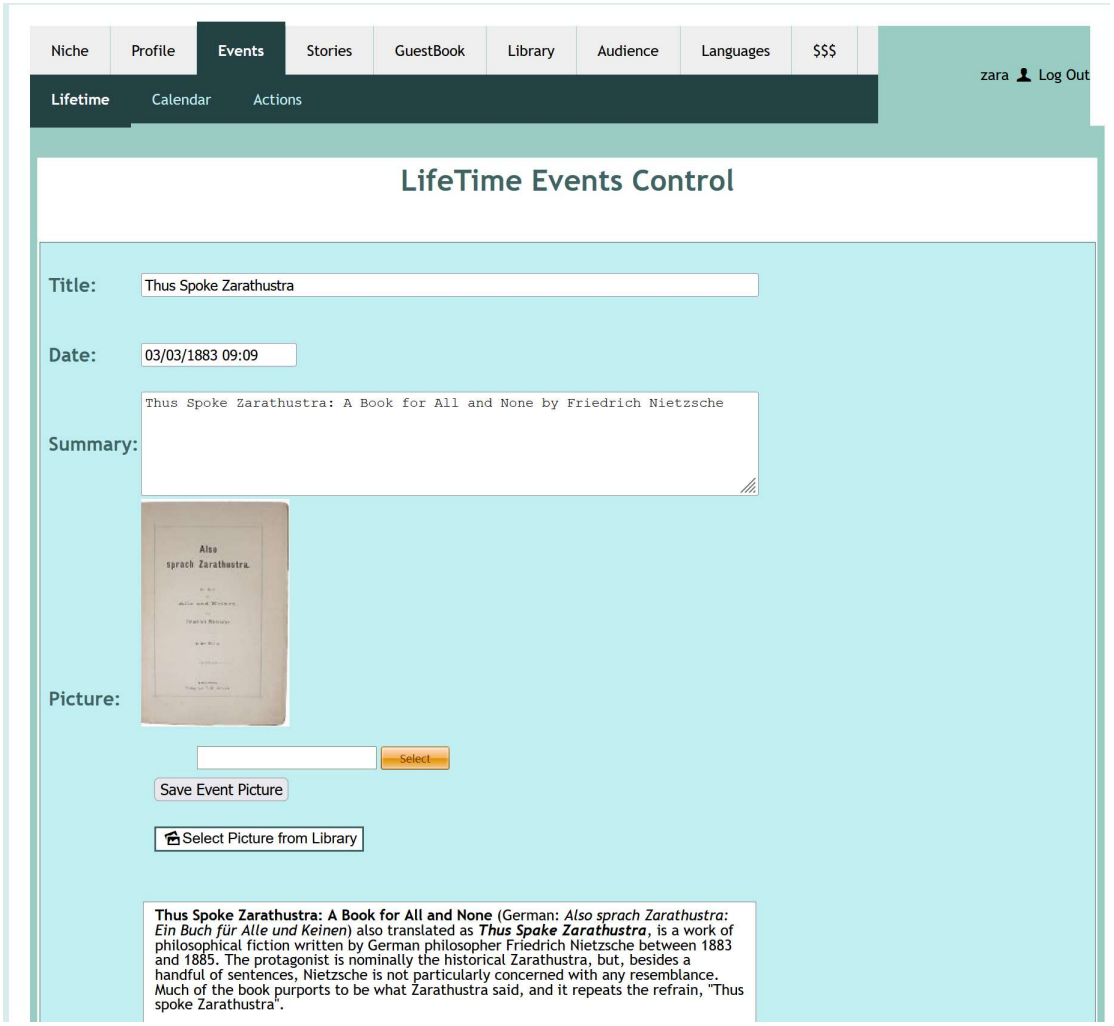


For our future blockchain operations we certainly will need some Ether, so we got some donated:

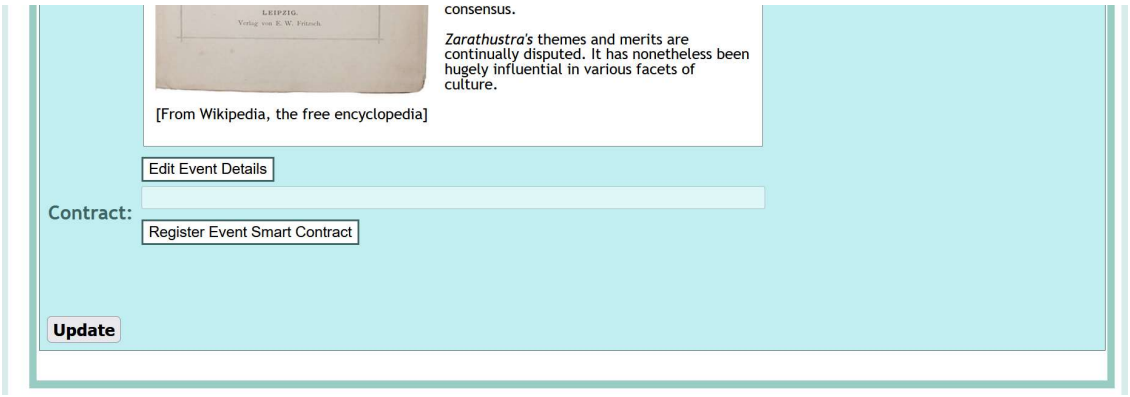
To add a new event, open “Lifetime Events” and complete the available fields:

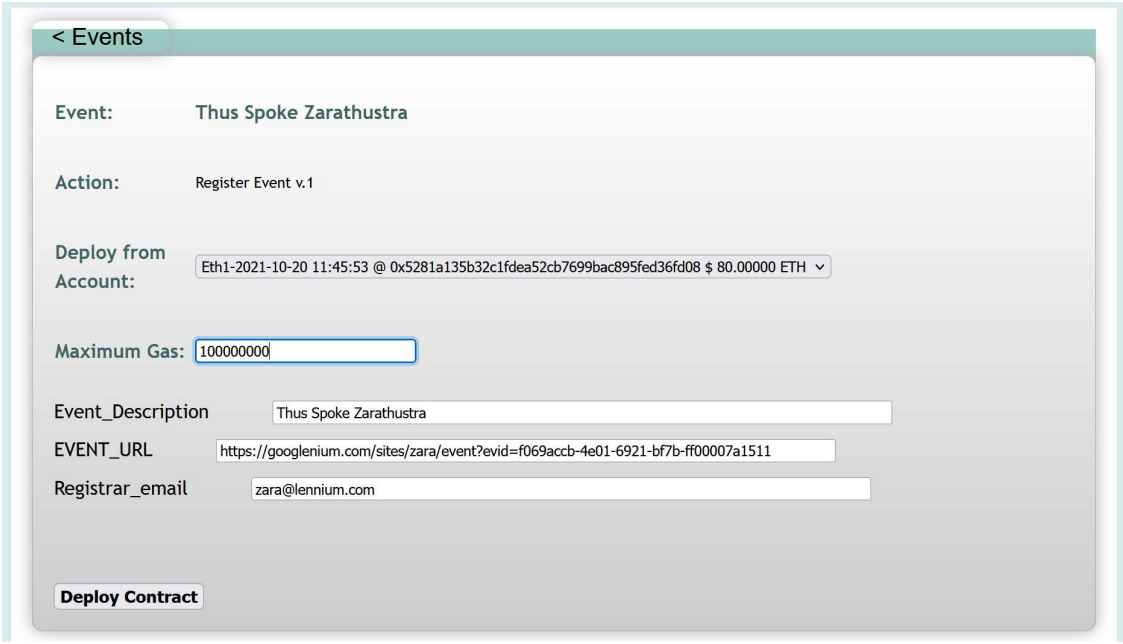




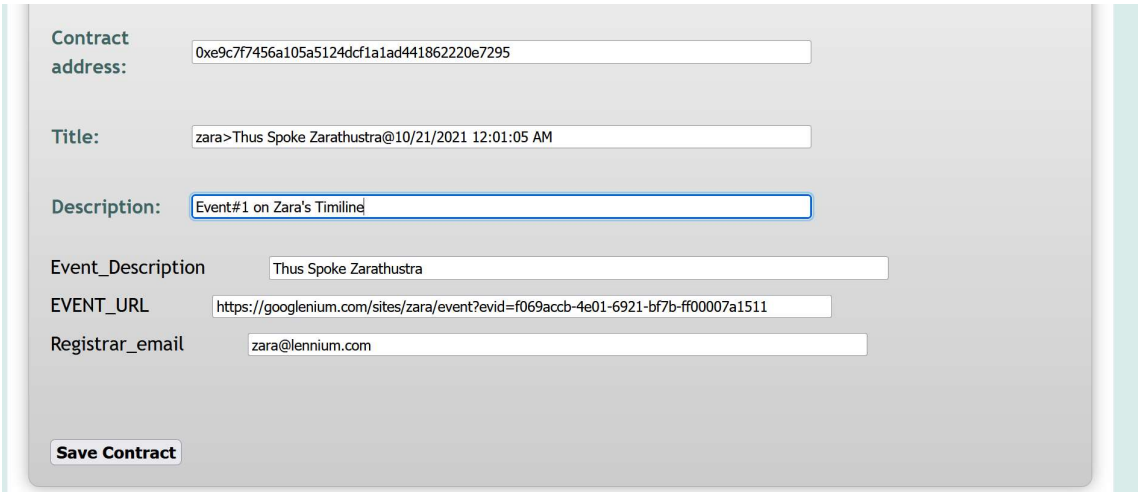


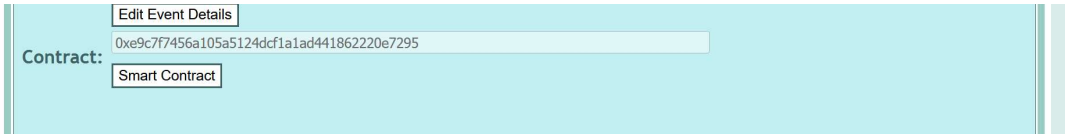
When all details are completed and saved, a newly created event can be registered on the blockchain:



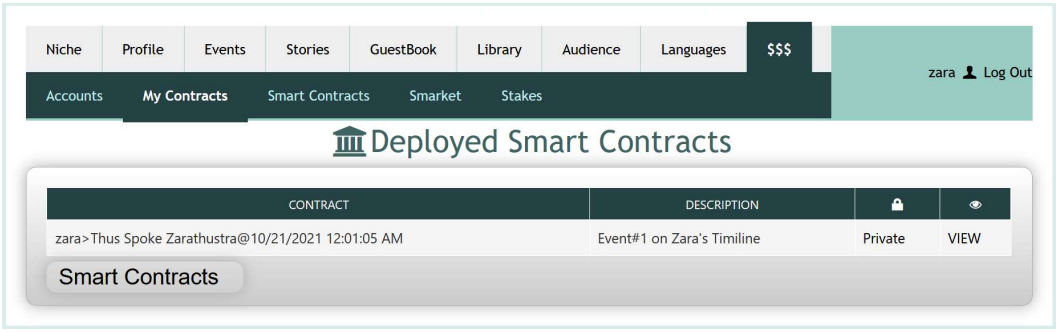


After deploying a Contract on the blockchain we need to save it in order for the ESC to be linked to the Event record:



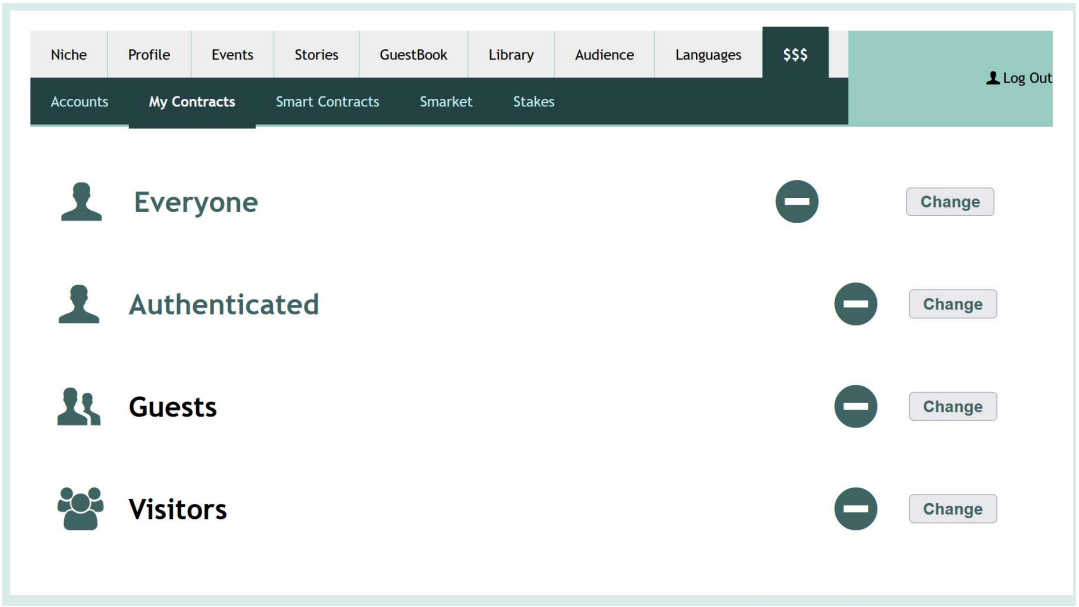


Now we have a new Contract in ‘My Contracts’ view:

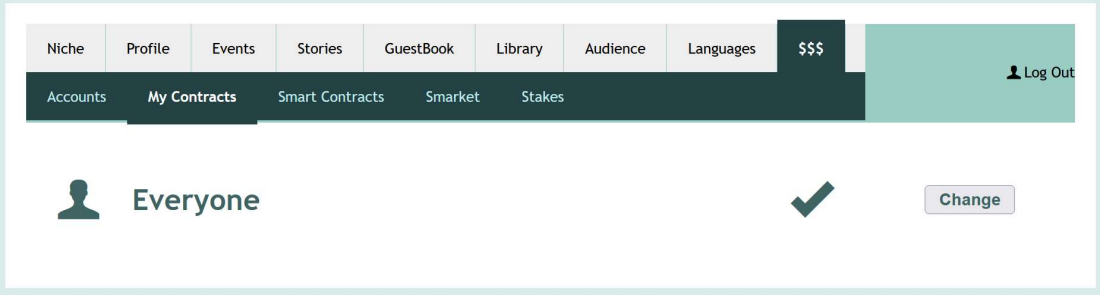


but it still has ‘Private’ only type of access.

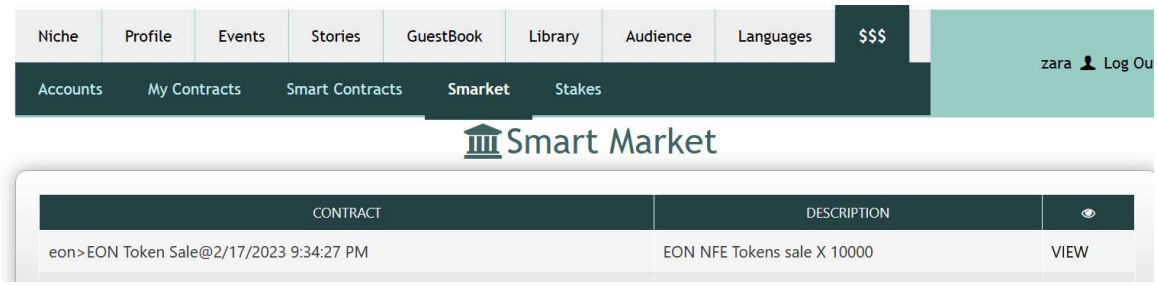
To make it visible to others we need to change its access level (click on ‘Private’):



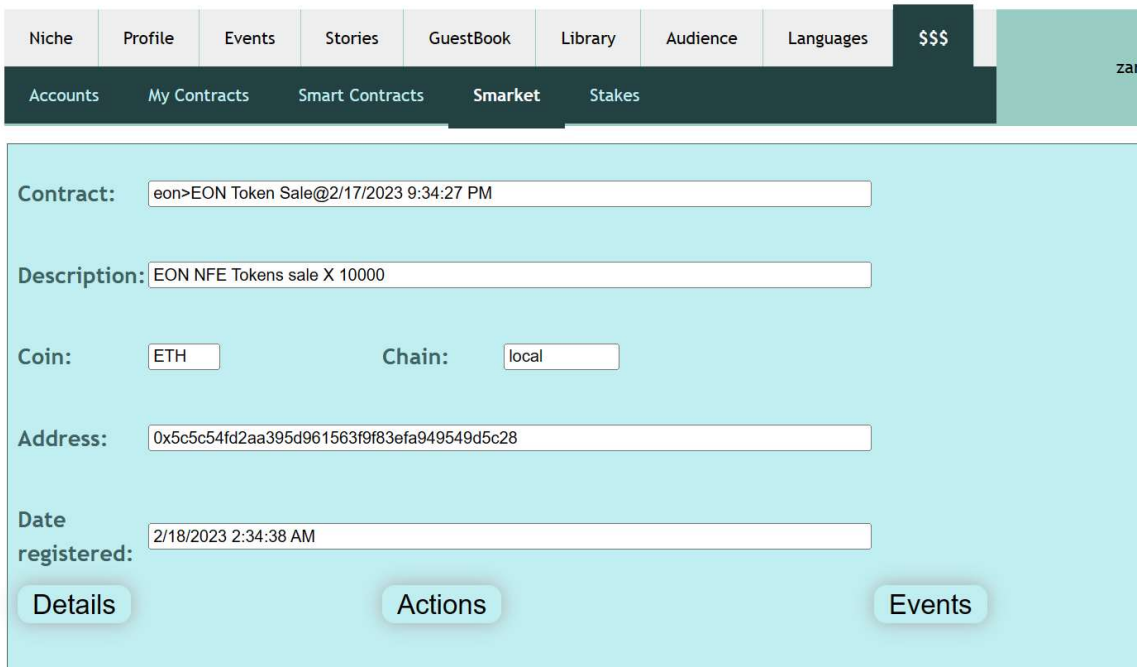
And make it visible for everyone:



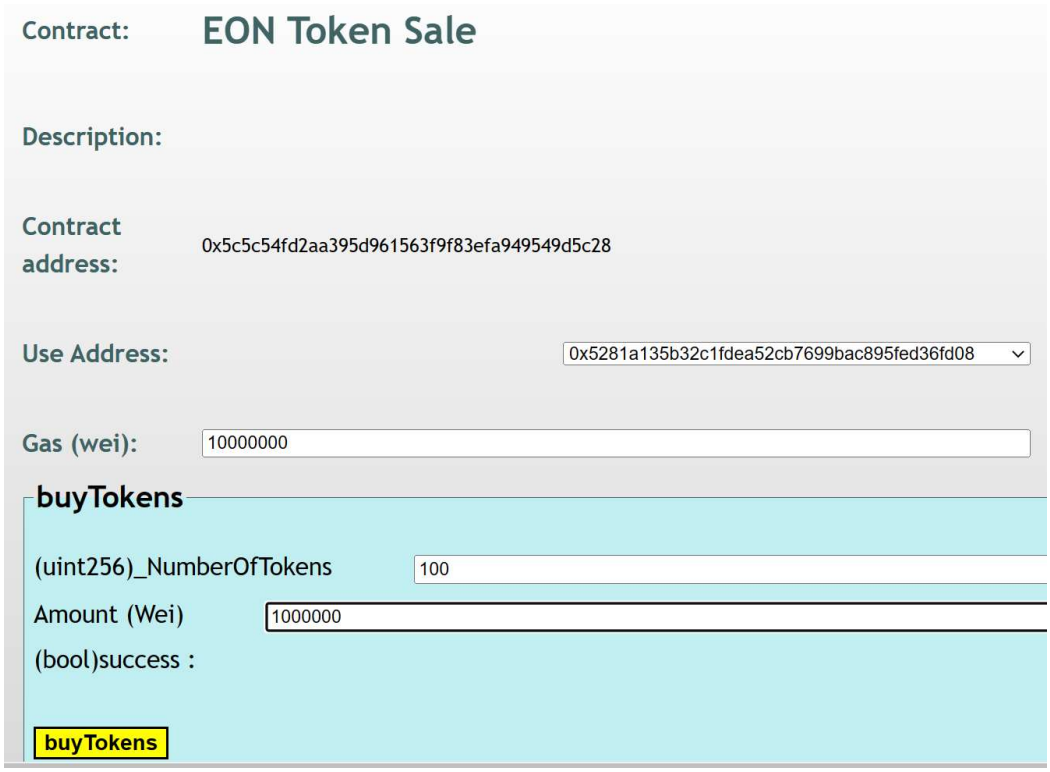
Next step would be raising new event stake by backing it with some EON tokens but we would need to get some tokens first. To do so we will go to the Smart Market “SMarket” tab and “View” a token sale smart contract:



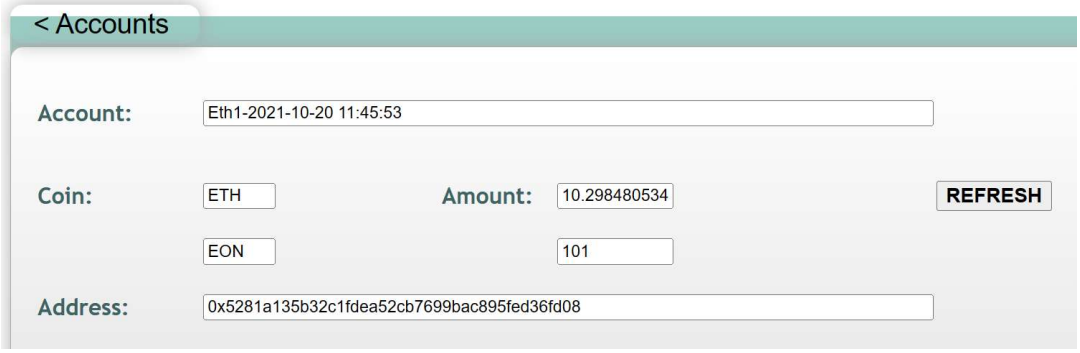
While in the contract, we select “Actions”



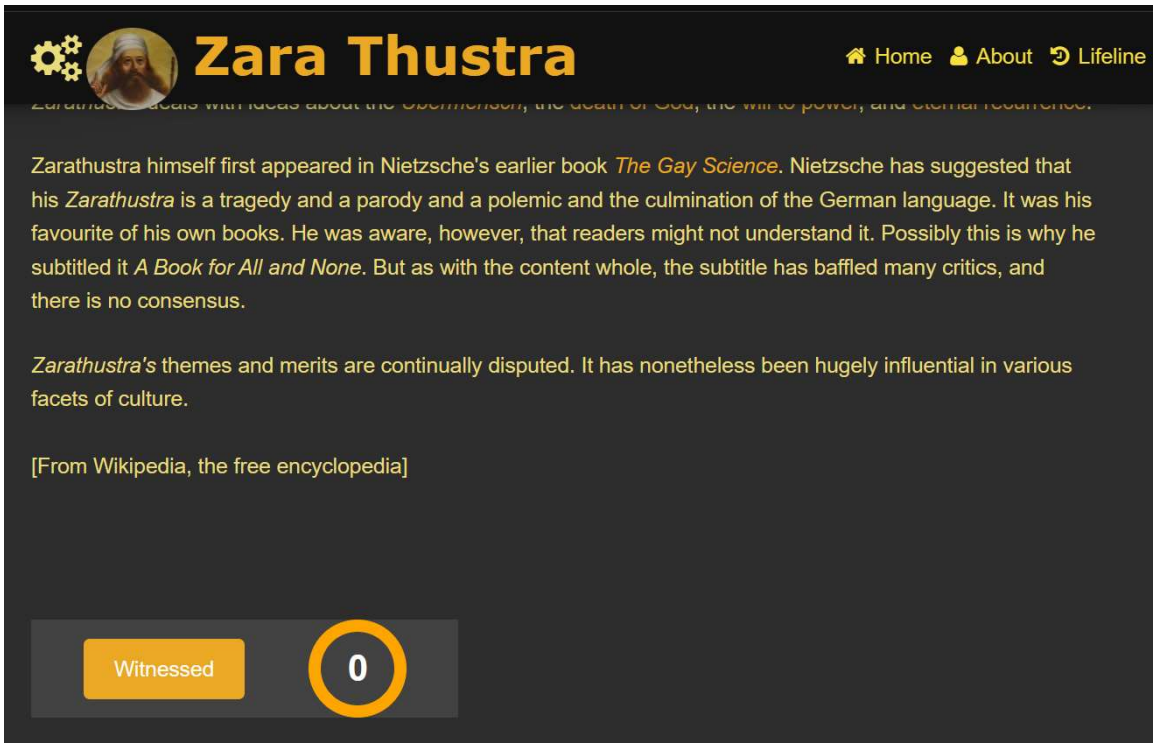
And exchange 1000000 Wei for 100 EON tokens – the rate set in the token sale contract.



The amount will be available in the account after “Refreshing” account screen:

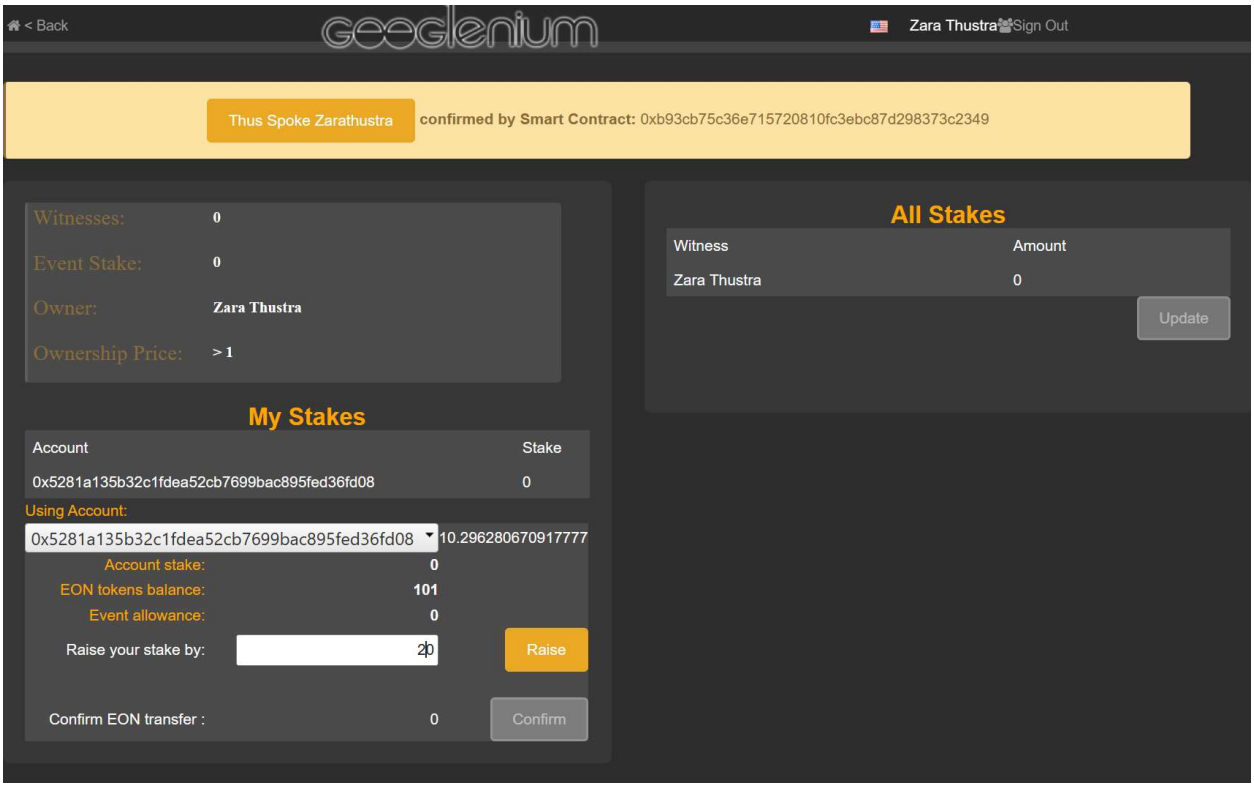


Now we can return back to the Event to raise the stake:

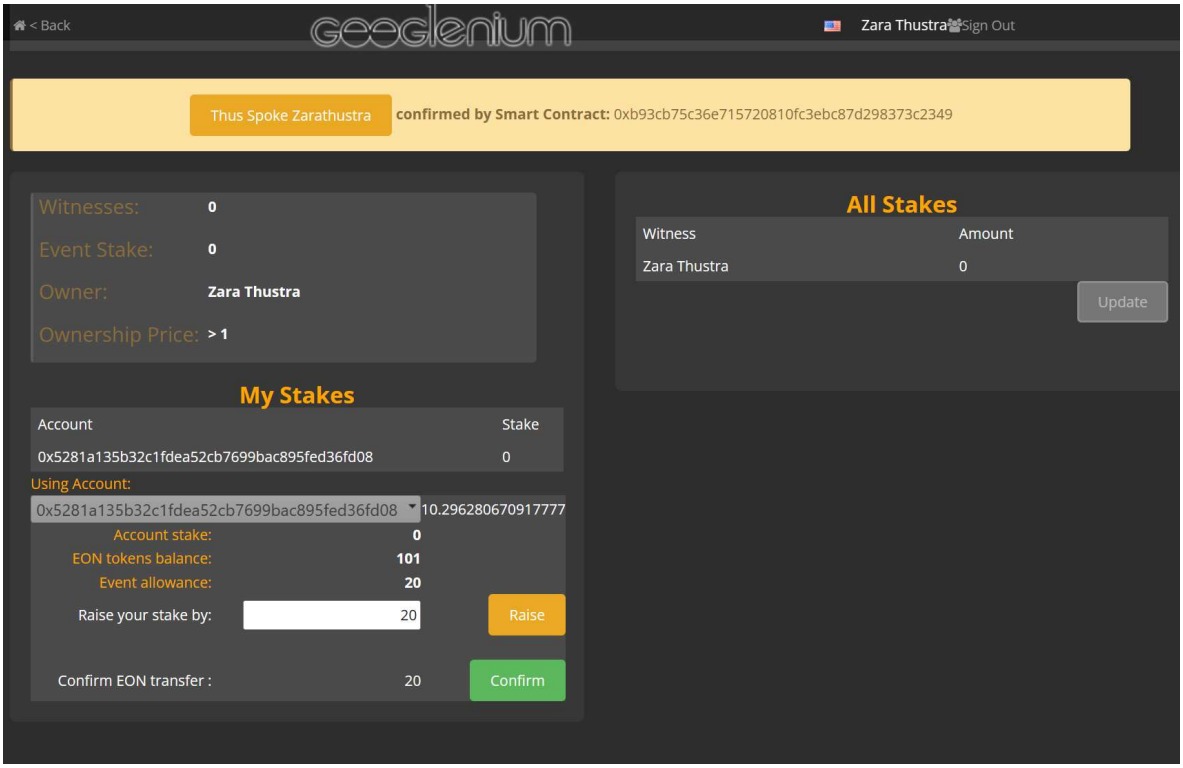


Operation is done in three steps:

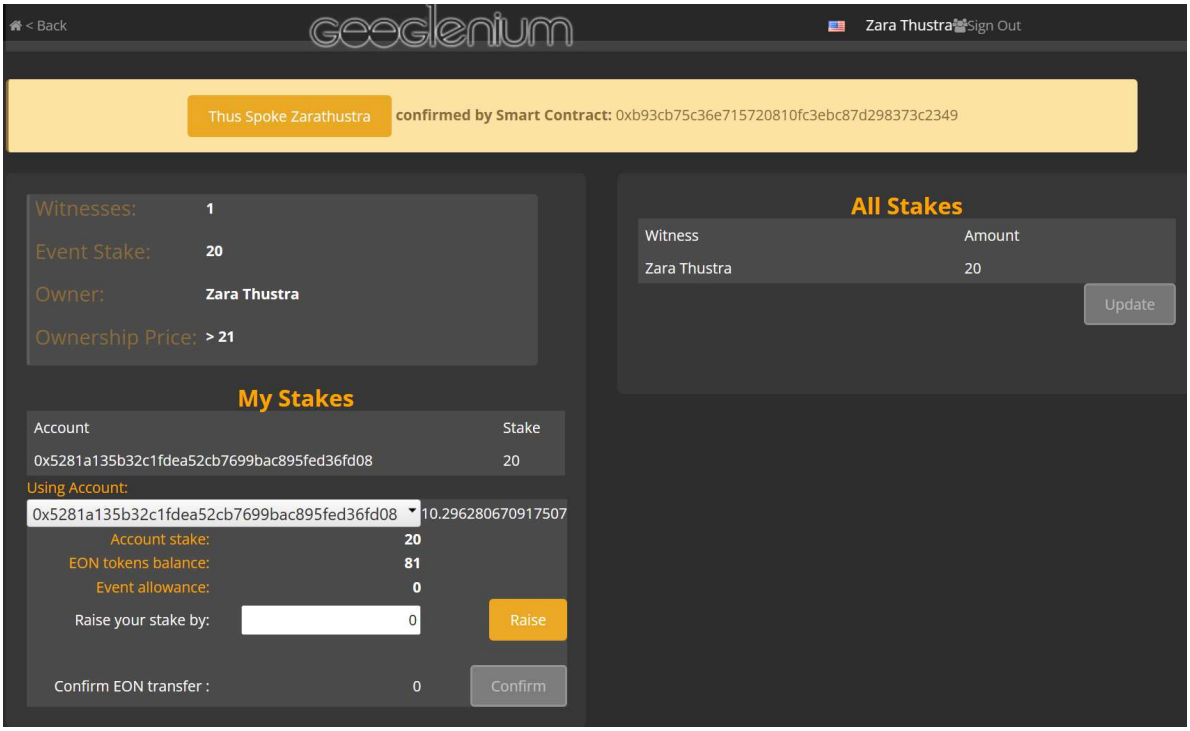
1. Securing amount from the Token contract (setting Allowance amount for the Event) – 'Raise' button:



2. Sending secured amount to the Event Contract ('Confirm' button):

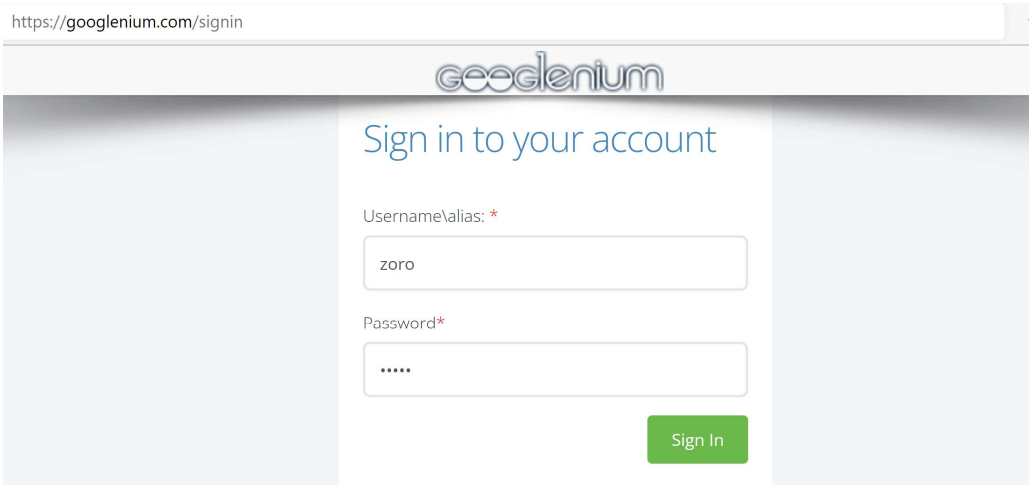


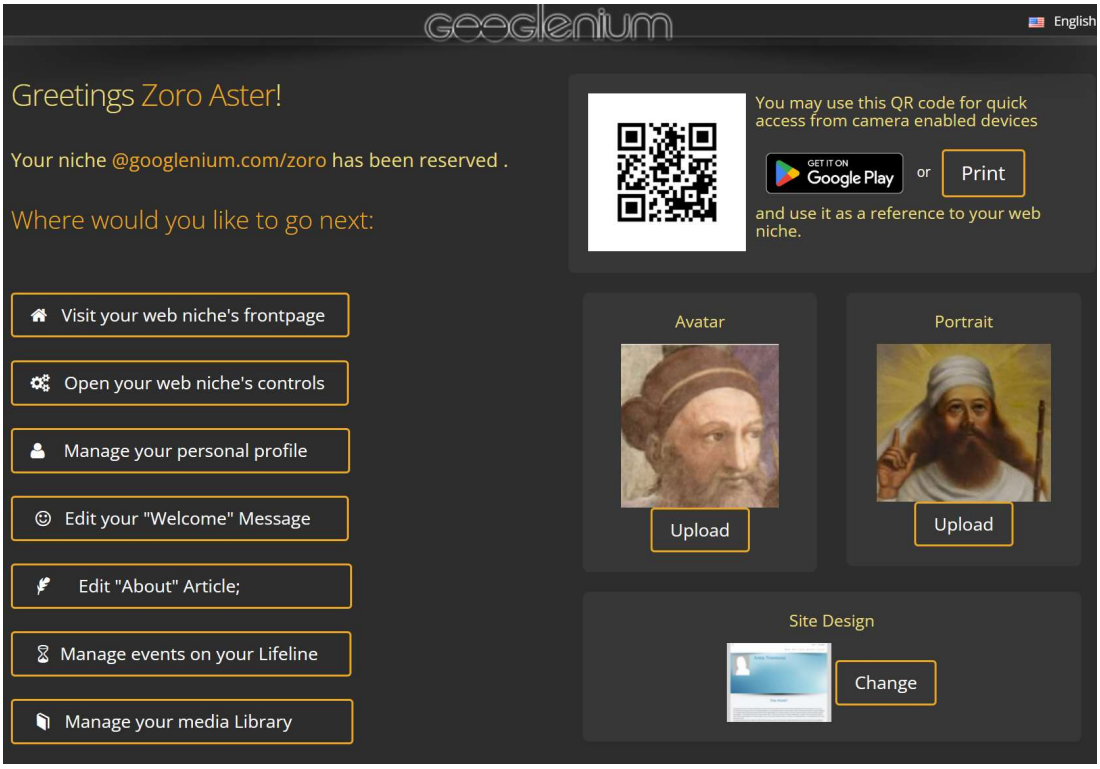
3. And updating Contract information in Googlenium database using ‘Update’ button:



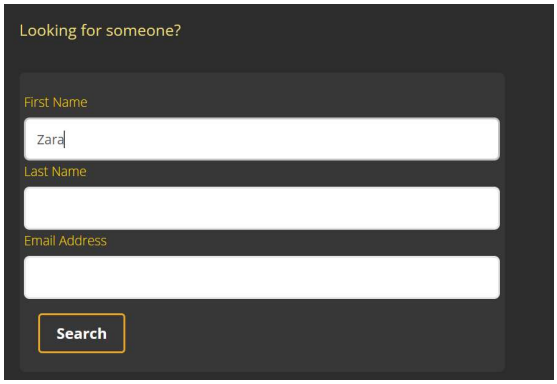
Other registered users, including Zara Thustra's another reincarnation – Zoro Aster, can also 'witness' the Event and raise their stakes and even take Event contract 'ownership' by staking amount higher than 'Ownership Price'.

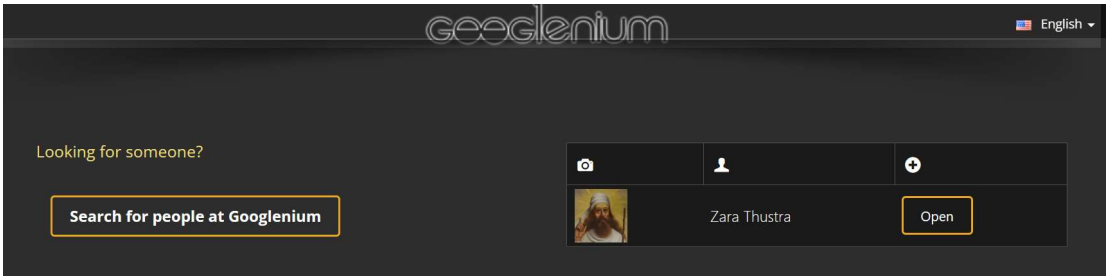
Sign in

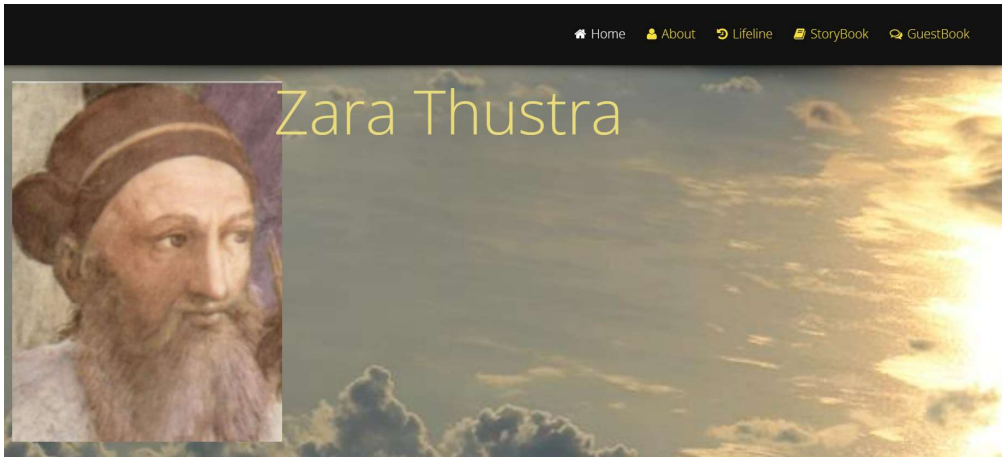




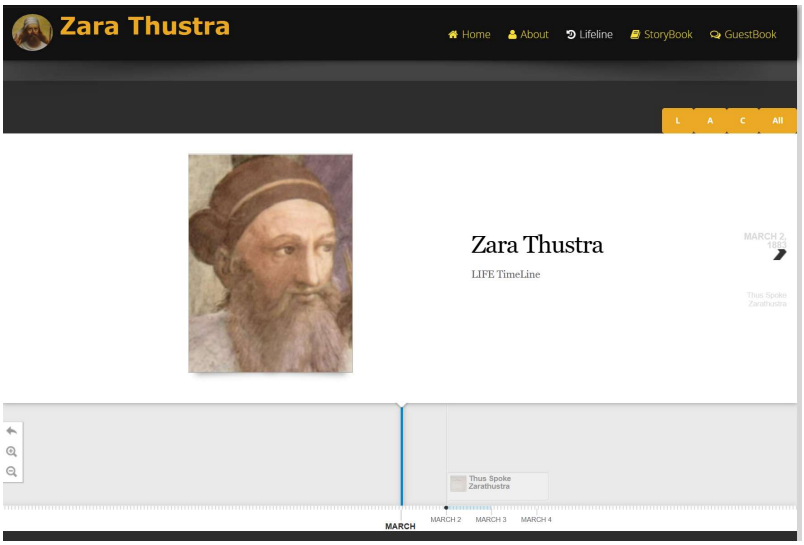
Click on GOOGLENIUM banner on top of the page and search for people:



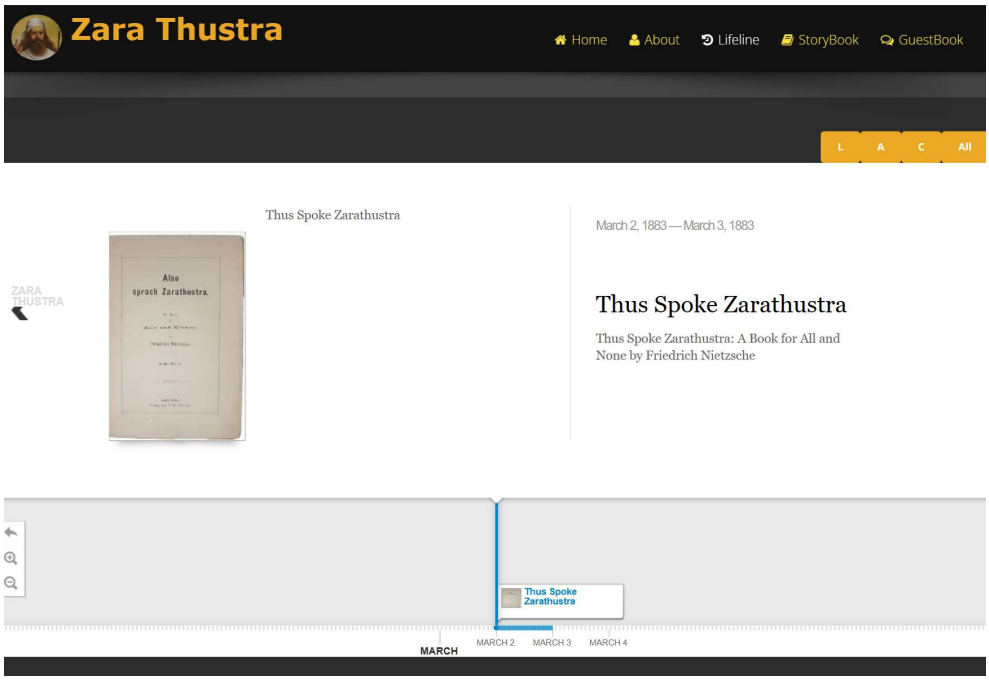




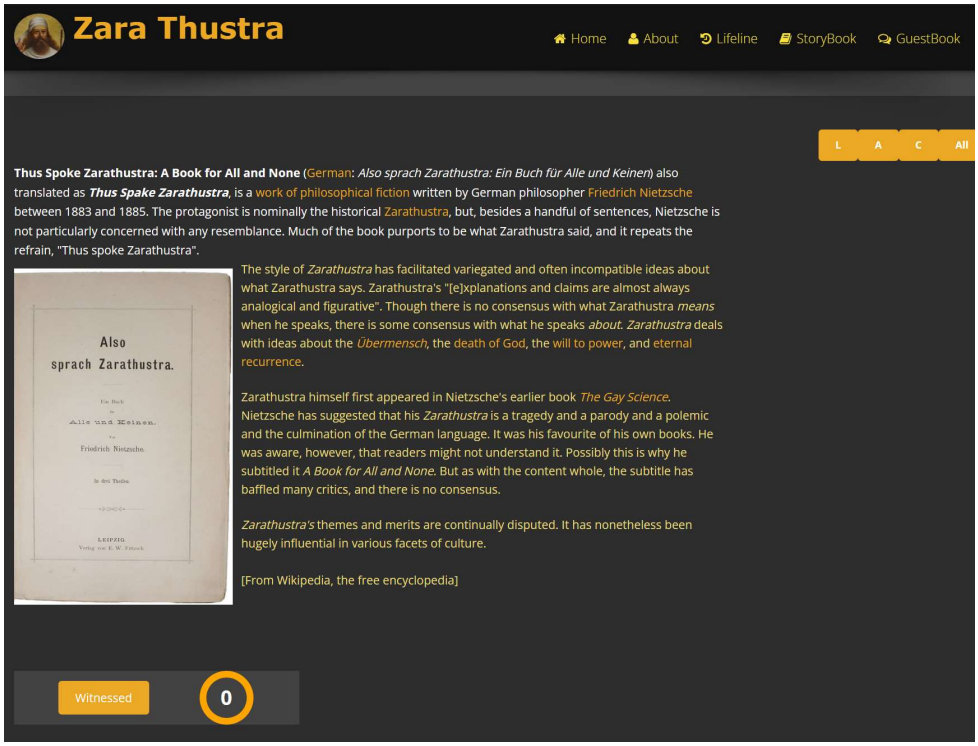
While on Zara Thustra's home page, go to "Lifeline:



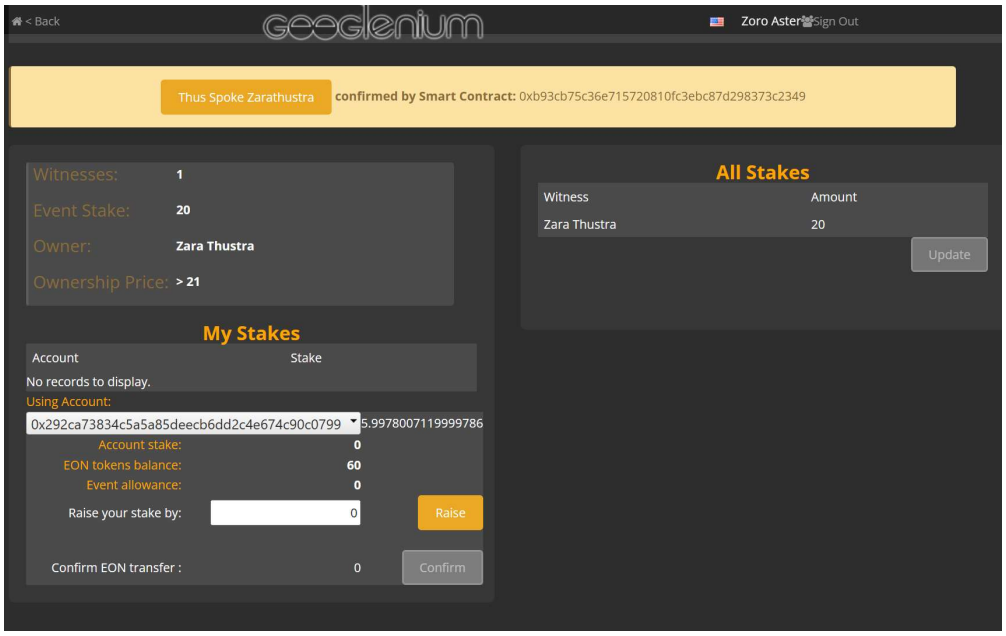
And select an event:



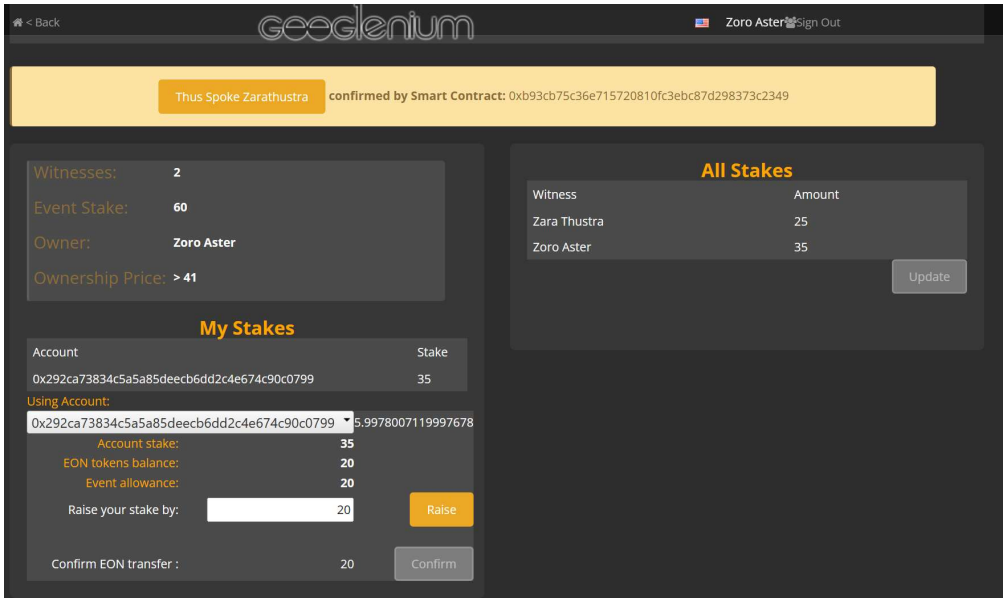
Click on the event thumbnail to see details.



Click ‘Witnessed’ button see current witnesses and their stakes:

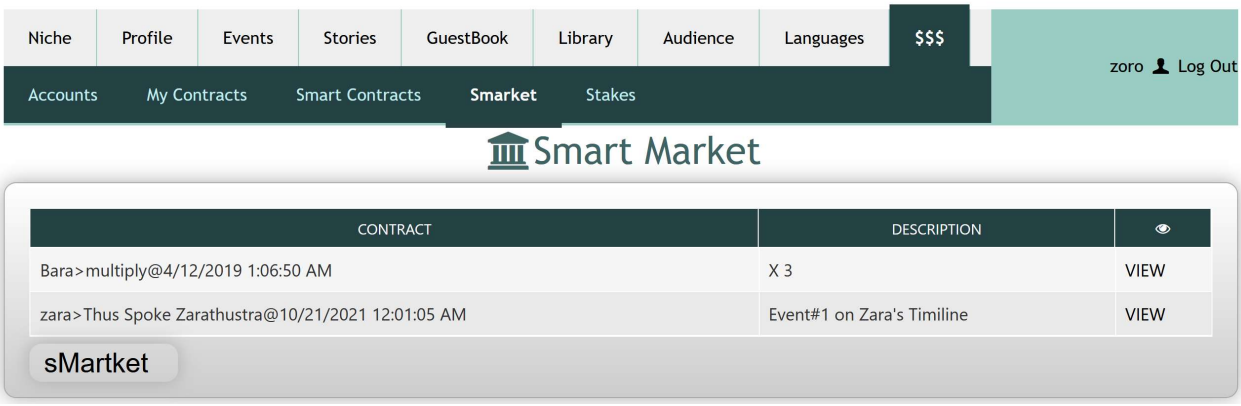


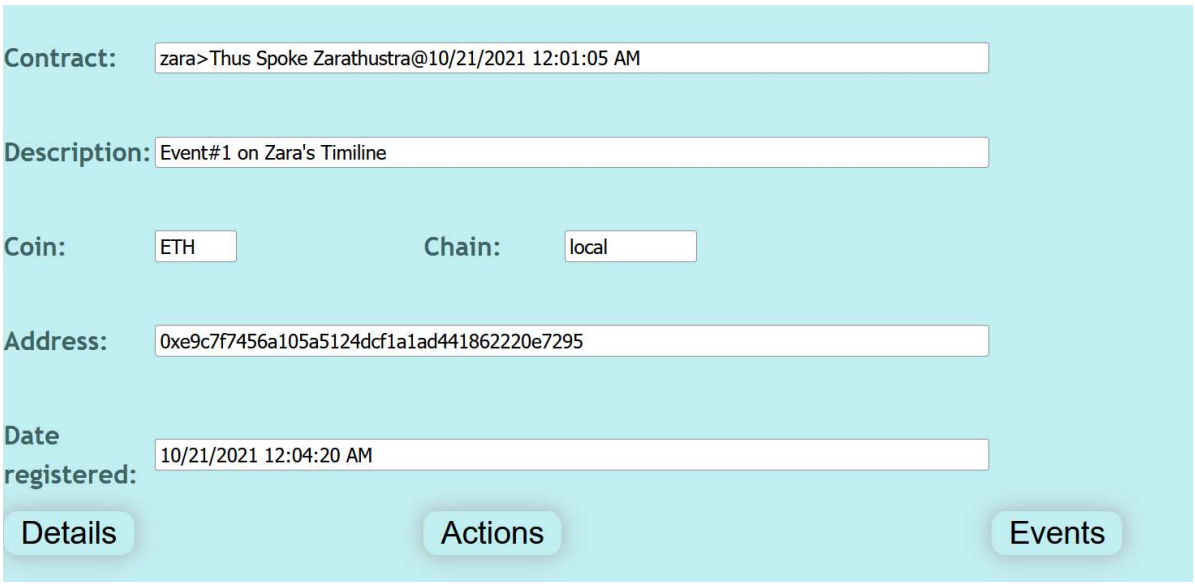
You can witness the event by adding any amount of EON tokens or become event owner by staking a sum larger than ‘Ownership Price’ in three steps operation described earlier.



All Event Contract details as well as Actions and logs are available in Controls ‘$$$’ section when Contract security level allows it.

Event Smart Contracts are located in Smarket (smart market) section:

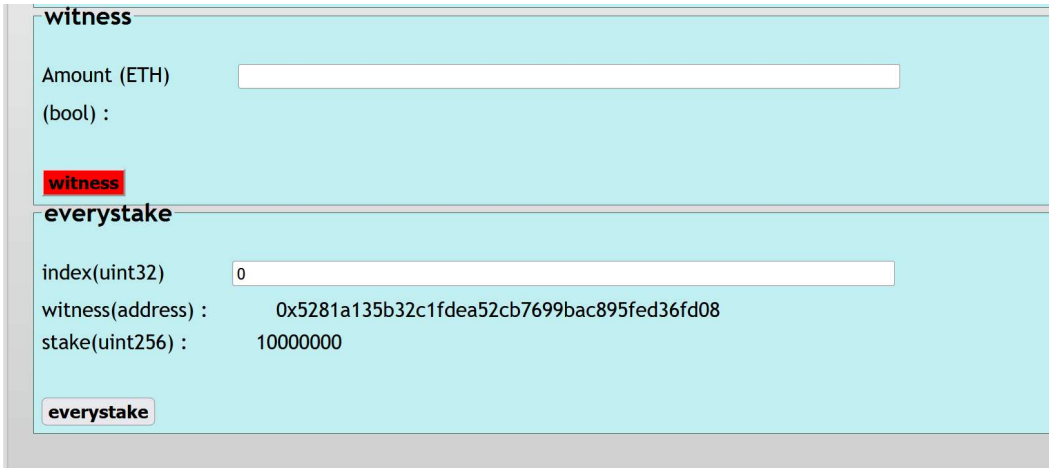




Using “Details”, “Actions” or “Events” buttons we can communicate directly with blockchain.

Details” will display current values pulling them from the smart contract:

"Actions" provide access to the contract functions:



A log of Contract events can be viewed directly on the blockchain in “Events”:

Log

Witness	Amount	WitnessStake	ContractStake	NewOwner	Time
0x5281a135b32c1fdea52cb7699bac895fed36fd08	10000000	10000000	10000000	True	10/21/2021 12:01:02 AM
0x292ca73834c5a5a85deecb6dd2c4e674c90c0799	25000000	19776813	25000000	True	10/31/2021 2:23:56 AM

Witnessed

Thank you for reading and please feel free to create your own account and play.

If you need some gas for operations on Googlenium local private Ethereum blockchain, please write to ***try@lennium.com***