BOUNTY PROGRAM AGREEMENT

PROJECT NAME

Treant - Actors Rendering

VERSION

1.0

TEAM

Miguel Celedón, Emmy Sandoval

ABSTRACT

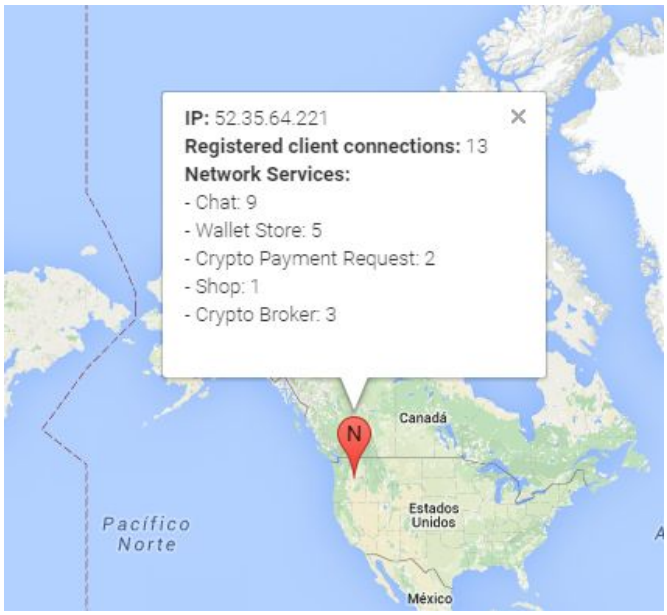
This project is aimed to improve the Treant site, with the ability of showing in the map the Actors and list the Network Services connected to a specific Node. We consider a **Node** as a server that works interconnecting clients between each other. **Clients** are considered as devices connected to at least one Node and contains Network Services and Actors. **Network Services** are connection points to a Node. Finally, **Actors** are special identities that runs on a devices but are treated as independent of those, in order to protect the pseudo-anonymity (Actors should not have a direct, public relationship between it and its user) of the end user that creates such Actor.

The alpha version will be available in http://lab.bitdubai.com/treant

SCOPE

1- List the Network Services connected to a Node and Client:

When clicking on a Client, the users will see a information window showing the list of the Network Services connected. The same occurs when users click on a Node, they will see an accumulative list of all the Network Services by type connected to it. The Client and Node markers will be different in design.

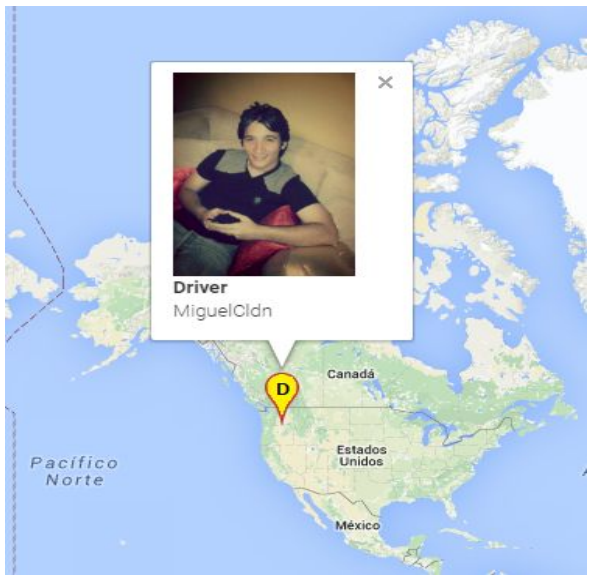


Network Services List - When the list is from a node, it will count the instances

Fermat components: N/A
Size: 20%

2- Show the actors present in the Network:

When clicking on a Node, users will be able, if they desire, to see the actors connected to such Node as special marker showing which Actor type is, and a line connecting them, with their profile pic and alias shown if the users clicks on any Actor. If no Node is selected, then all actors will be shown without connecting edges.



Draw Special markers and show the actor data when the user clicks such actor.

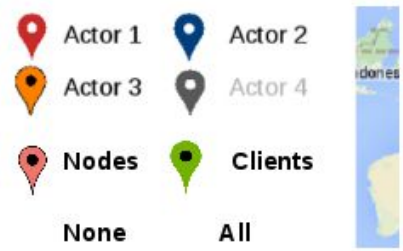
Note: As the number of possible Actors is in continuous growing, we will limit the development and design of the special markers to the ones listed below, still there is the possibility to add new markers and their data in a json file for the future:

- Chat
- Shop
- Crypto Broker
- Crypto Costumer
- Asset Issuer
- Asset User
- Artist
- Fan

Fermat components: N/A
Size: 50%

3- Filter the Actors markers by Actor type:

If users wants to only see one type of Actor in the map, the filter will have an option so they can show or hide any Actor type, or show none or all.



Here we see the actors filter, Disabled markers will be shown in gray like Actor 4

Fermat components: N/A
Size: 30%

EVALUATION

For the acceptance of this project, the user should be able to:

- Click on a Node and see its connected NS. The same applies for Clients.
- Click on a Node and then click on a button “Show Actors” and see the actors in the map, showing its alias and profile picture.
- Write or select an Actor type in the filter to show only those ones.

TERMS AND CONDITIONS:

1- The team agrees that the implementation project has two stages: functionality and beta testing.

2- The team understands and accepts that the functionality will be considered done when all the features described in the scope of this agreement are completed and tested in an alpha stage.

3- The team understand and accepts that the implementation of functionality must follow the Fermat's technical guidelines.

4- Component architecture and workflows are created in the Analytics System and the Interfaces in the API library of the platform involved.

5- The team understands and accepts that implementation will be evaluated by the @bounty-program-team and will include GUI/UX design checking, functionality test and review of Fermat's technical guidelines compliance.

6- The team understands and accepts that there is only one free review for functionality and one for Fermat's technical guidelines compliance. The following reviews will cost the team 25% of the related bounty each in case the first one wasn't approved.

7- The team agrees to complete the implementation on the following conditions:

- **Implementation bounty:** The functionality will be 70% of the total bounty. This bounty will be awarded to the development team when the @bounty-program-team considers that the functionality delivered is done.

- **Implementation due date:** All the features will be finished before **07/06/2016**. If on this date the development team does not deliver the agreed functionality, it will lose the implementation bounty.

- **Implementation collateral deposit:** The team agrees to deposit the amount of **$1.500** paid in tokens in favor of the Fermat Foundation, as a collateral to be lost if this part project is not approved before the due date.

- **Implementation margin:** No more penalties apply 7 calendar days after implementation due date.

- **Implementation penalty: 10%** of the implementation bounty for each calendar day that elapses after the implementation margin without formal acceptance from the @bounty-program-team. This penalty will be paid by the development team from its savings to the Fermat Foundation. If savings are not enough it will be deducted from their cash salaries.

8- The team understands and accepts that beta testing will be conducted by the @beta-testing-team.

9- The team understands and accepts that criteria to pass beta testing are:

A. No bug issues on beta testing due date,
B. Or no bug issues in a period of three (3) consecutive calendar days before the due date,
C. Or no answer from @beta-testing-team about solved bugs for (6) consecutive days before the due date.

10- The team agrees to complete the beta testing on the following conditions:

- **Beta testing bounty:** The beta testing bounty will be a fixed 30% of the total bounty. It could be awarded to the development team if it passes the beta testing on time or by @beta-testing-team if it fails. Also, it implies that development team will not get this bounty unless it succeeds in the beta testing process.
- **Beta testing due date:** Beta testing will be passed before **20/06/2016**. If on this date the development team does not pass the beta testing, it will lose the beta testing bounty, which will be automatically awarded to the @beta-testing-team.

- **Beta testing collateral deposit:** The team agrees to deposit the amount of **$1.500** in Fermat tokens in favor of the Fermat Foundation, as a collateral to be lost if this part of the project is not approved before the due date.

- **Beta testing margin:** No more penalties are apply 5 calendar days after the beta testing due date.

- **Beta testing penalty: 10%** of the bounty for each calendar day that elapses after the due date without formal passing through beta testing. This penalty will be paid by the development team from the implementation bounty previously awarded or its savings to the beta testing team. If savings are not enough it will be deducted from their cash salaries.

TOTAL BOUNTY

The total amount of the bounty in Fermat tokens for this project is **$5.000** ($3.500 for implementation + $1.500 for beta testing).

DISTRIBUTION OF BOUNTY BY CONTRIBUTOR

Emmy Sandoval 15%
Miguel Celedon 85%
Total amount deposit: $3.000

SUMMARY

Implementation due date	**07/06/2016**

Implementation collateral	**$1.500,00**
Implementation margin (days)	**7**
Implementation penalty (%/day)	**10**
Functionality review (attempts-25%)	**1**
Technical review (attempts-25%)	**1**
Implementation bounty ($)	**$3.500,00**
Beta testing due date	**20/06/2016**
Beta testing collateral ($)	**$1.500,00**
Beta testing margin (days)	**5**
Beta testing penalty (%/day)	**10**
Beta testing bounty ($)	**$1.500,00**
Total bounty ($)	**$5.000,00**