bEquality - Automated tracker for gender equality with blockchain Creation of an automated and reliable gender equality index

Lena Csomor, Qianchen Yu, Lena Csomor, Noah Berner, Eric Léger,
Fredin Thazhathukunnel, Matthias Matti, Lino Telschow, Roy Schubiger

February 18, 2018

This project has been built in the context of the BIOTS Hackathon 2018
Blockchain and the Internet of Things School

Project repository: `https://github.com/ETHBiots2018/bEquality`

Abstract

Today, most companies' market value is driven by intangible value,such as reutation or branch equity. Thus, thousands of companies now provide detailed extra-financial information. Regulations are mandating increased management disclosure and analysis on sustainability, and investors are analyzing the comprehensive risks and opportunities of issuers in public and private markets. One aspect of this extra-financial information is data on gender equality.
And more and more investors take this data into consideration when investing as research suggests that good gender equality practice can serve as an indicator for good corporate governance and decision making and these companies might outperform in the future. To get a structured, concise and comparable overview of this data, investors often relay on established gender-equality certifications. Unfortunately, this process is costly, time intensive and requires multiple revisions to produce reliable data. [1]

Our project aims to simplify and automate the process of obtaining such a gender-equality certifications based on an existing Gender-Equality-Framework, and to make its results publicly accesible for everybody.
Our approach focuses on the data capture, data storage, data validation and a the display solution of the process.
The data is obtained by the means of a website and an app whereas the storage solution relays on an E-voting system, based on the latest blockchain and cryptographic technology. The solution provides the code of the communication interfaces for the website and the app, as well as the necessary code to handle the data for the E-Voting system via blockchain.
Furthermore, a Radar-Chart is used to present the results in a straightforward manner.

In conclusion the platform ensures a trustful, transparent and cheap way to create a gender-equality-index, which is useful for people to make investment decisions and for companies to further improve their status with respect to gender equality.

[1] This paragraph is taken from the challenge description handed to us by the UBS

Introduction

The current situation of the economy is not fully inclusive for every member of the society, women get discriminated, sexually harassed and do not have equal chance of high salaries and or higher positions in companies.
Our Project bEquality aims to punctually recognize inequality by providing a transparent, reliable and efficient gender-equality index for companies. We think that applying and evaluating a gender-equality index should not be expensive and exclusive to companies that can afford it.
We aim to improve todays state of society with respect to gender-equality in the economy.

This document summarizes our approach to solve this problem. Our solution has come up on a hackathon organized by the Blockchain and Internet of Things School (BIOTS) 2018.

We will first present you our whole process systematically and then later dive further into the implementation of the whole process.
The reader should have basic programming understanding or basic logical thinking to understand our approach and implementation, we try to explain it as easy as possible.

-describe structure of report - done
-describe prerequisities - done
-general things - Q: what to add more?

1 Project

1.1 Overview

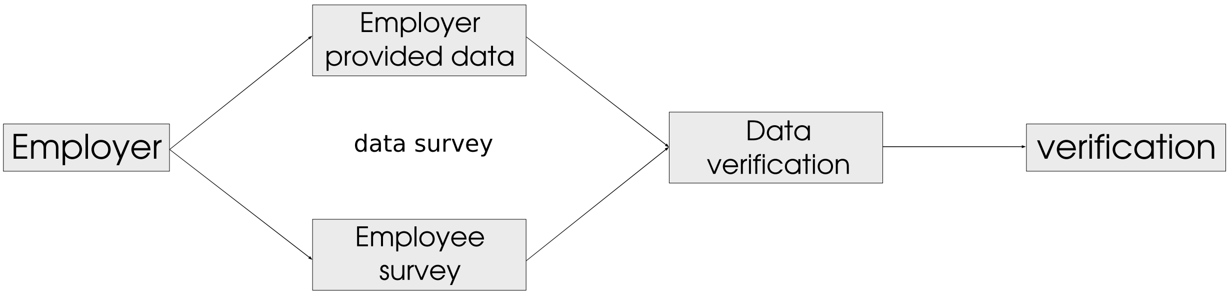

Figure 1: Project flow graph

The figure 1 visualizes the high-level structure of our project.
Existing gender-equality indices differ from our project at the points *data survey (employee survey)* and *data verification* in the high-level structure and in the final evaulation.
We will now explain the high-level structure of our project structure.

Employer Survey

There are a few necessary steps for an employer to get a certification. First, the employer has to to send the data asked for by the given rating framework via a Website. Additionaly the company provides the E-mail-Addresses of their employees. That's the employer provided data.

Question: do we really take the e-mail addresses?
Or does the employer tells every employee to make an account on our platform and then the employer sends us all generated public keys? (sending of public keys can be done implicitly at the registration of the employee)

Employee Survey

After that, a percentage of employees will be chosen randomly to participate in an employee survey. This survey is used to validate the plausability of the provided data from the employer and to obtain further important private data from employees that cannot be known by the employer.

However, this is an easy task for the employee. To set up an account, the employee gets an invitation link from our program and then gets linked to the app, where he/she can set up his/her account. While setting up the account, a public and a private key get generated in the background and linked to the employee account.
After the account set-up the company sends all the public-keys generated by the employee account set-up to our framework, this then initiates the survey set-up.
When all background work is done, the employee gets a message that the survey is ready. He/she has then just to log into the app, fill out the survey and then submit the data.

Everything technical is handled automatically in the background, such that the user just observes a handy interface, and all data is stored on either the blockchain or the IPFS.

Data Validation

When all data is gathered from all chosen employees, we split the data into different classes. Either the data from the employees intersect the employer provided data, and is therefore in the *intersecting-data category*, or it is data that is just known or provided by the employee (such as number of sexual harassments or equivalent things).

All data from the *intersecting-data category* gets compared by comparing the provided view of the employer and the view of the employee.
This comparison then provides a validity factor for the employer provided data.

This process of data validation ensures that the data is not corrupted by either humans or bots.

Q: How is validation done exactly?

Display Solution/Evaluation

We would adapt our evaluation to already existing gender-equality indices. But to further improve the evaluation of this evaluation, we would like to evaluate also different indicators for gender-equality and other measures that are important.
And because all the data is digitally available, we could also insert classification via artificial intelligence to even further improve the evaluation process.
The data evaluation then finally not just results in a single index number, but in a multidimensional spider-diagram, that allows the investor or the company to look closer to existing problems or advantages over the requirement:

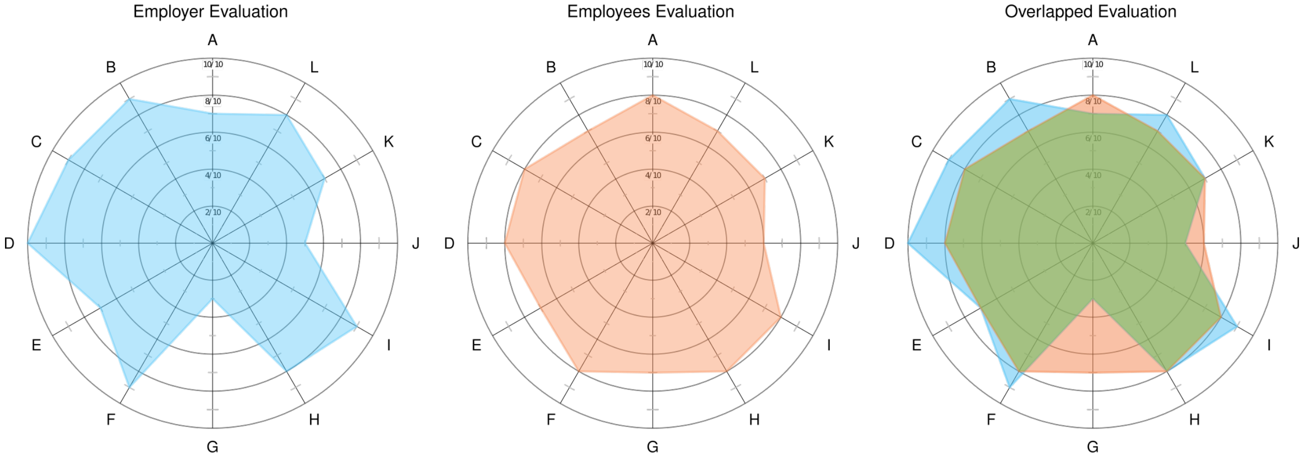

Figure 2: Spider-diagram evaluation example

The figure 2 provides a flexible evaluation system with detailed feedback.
This detailed feedback is useful for company management and also useful for investment decisions by investors that may want to invest in this company. It also implies a feedback system for the company to rate, such that the company knows what to improve in it's employee environment to further imrove it's index.

1.2 Challenges to Solve

There are still some challenges to solve for this project.

One major challenge is to provide a secure way of implementing a system equivalent to E-voting, i.e. to find a system where a user applies to a survey or evaluation and the data provided by him cannot be traced back to this user, even when the user can verify that the data provided by him is used and not some modified version of his data.
A way of solving this problem for this use-case (of a gender-equality index) it may be a solution to implement the system in such a way, that cheating for companies are too expensive to be profitable and the company therefore acts cooperatively and provides true data.

challenges for implementation
-Finding a secure way of implementing a system equivalent to E-voting.
-Making cheating for companies too hard to be profitable.
-privacy for gathering data, boss should not have possibility to see results from its employees
-costs

done???

2 Technical Implementation

2.1 Data Capture and Storage

As already mentioned earlier, the *data capture* process is split into two parts.
The first part is the evaluation of a survey handed to the management of the company, i.e. the *company provided data*. This first part can be implemented via a survey that is being transmitted to us.
The second part of *data capture*, the *employee survey*, is a special point of interest. Because it is important to guarantee anonymity to the employee and at the same time ensure validity and transparency of the data that is provided. We implemented this second part with a protocol that is equivalent to an *E-voting system* that is explained in detail below.

E-voting protocol

A company that wants to be part of our rating applies and sends their unverified data via smart contract (contract `SurveyFactory`) to the blockchain. This initiates the whole process of survey generation.

```
// https://github.com/EriCreator/bEquality/blob/master/Survey.sol
/* SurveyFactory serves as a hub
    (deployed on the blockchain upon the launching of bEquality)
    Company can create their own survey by providing a list of permitted user address. */

contract SurveyFactory {

  //address owner;
  mapping(uint => address) public SurveyContracts;

  // function SurveyFactory(address adr) public
  // {
  //  owner = adr;
  // }

  function createNewSurvey(uint companyID, address[] addressessOfEmployees, string
      _hashToaddressessOfEmployees) public returns(address newContract)
  {
    // require(msg.sender == owner);
    require(SurveyContracts[companyID] == 0x0);
    Survey c = new Survey(addressessOfEmployees, _hashToaddressessOfEmployees);
    SurveyContracts[companyID] = c;
    return c;
  }

  function getContractAddress(uint companyID) public constant returns (address)
  {
    return SurveyContracts[companyID];
  }
}
```

They also send bEquality the email addresses of all emplyees, as observable in the function `createNewSurvey`), which we store in a secure and private data base.

We do not store the password of the employee explicitly, we instead store the hashed password linked to the corresponding username. To further link the private key to the account, we also store the hashed private key on our database.

This account stored on our database does not allow us to login to the user-account, but we can verify validity of the account, when a user wants to log in.

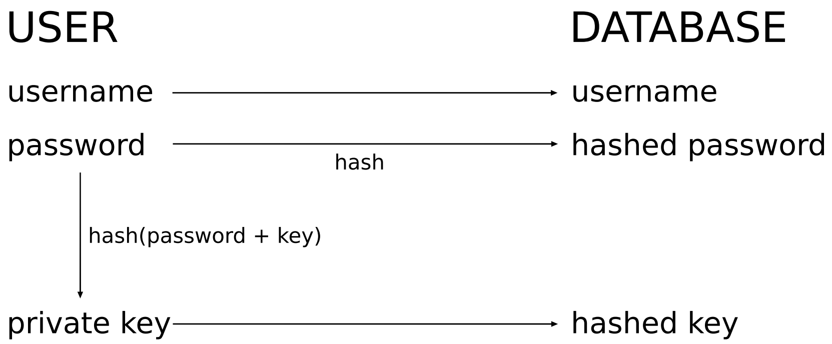

Figure 3: Secure login data storage

Is this correct?

When all accounts are set up and put money on them to pay for the transaction costs, the employee gets notified that he has to log into his account and fill out the survey.

The employee then acts on the following smart contract `Survey`, that only he can modify, but not the contract-creator. When the employee has filled out his survey, he submits his data via a user-friendly interface and the data gets stored anonymously on the blockchain. The user therefore can verify that his data is not altered, but at the same time there is guaranteed that the employer cannot prosecute the employee for telling the truth about the company.

```
// https://github.com/EriCreator/bEquality/blob/master/Survey.sol
/*
   Survey is the child contract created by the SurveyFactory where only the permitted user can modify.
*/

contract Survey {
    mapping (address => string) public hashes;
    mapping (address => bool) public isAllowedToSumbitSurvey;
    string hashToaddressessOfEmployees;

    function Survey(address[] addressessOfEmployees, string _hashToaddressessOfEmployees) public {
        hashToaddressessOfEmployees = _hashToaddressessOfEmployees;
        for (uint256 index = 0; index < addressessOfEmployees.length; index++) {
            isAllowedToSumbitSurvey[addressessOfEmployees[index]] = true;
        }
    }

    function submitResults(string myHash) public {
        require(bytes(hashes[msg.sender]).length == 0);
        require(isAllowedToSumbitSurvey[msg.sender]);
        hashes[msg.sender] = myHash;
    }

}
```

Because all data is stored on the blockchain, everyone can verify the integrity of our analysis, and can even make his own analysis of the given data. This allows full transparency towards the shareholders and future investors in the company.

Yet there are still challenges ahead of us, for example the possibility of storing the Ethereum addresses on the blockchain instead of a private database for further transparency and automation of the process.

Below is shown the full protocol in a schematic way, to give the reader an overeview of our E-voting protocol.

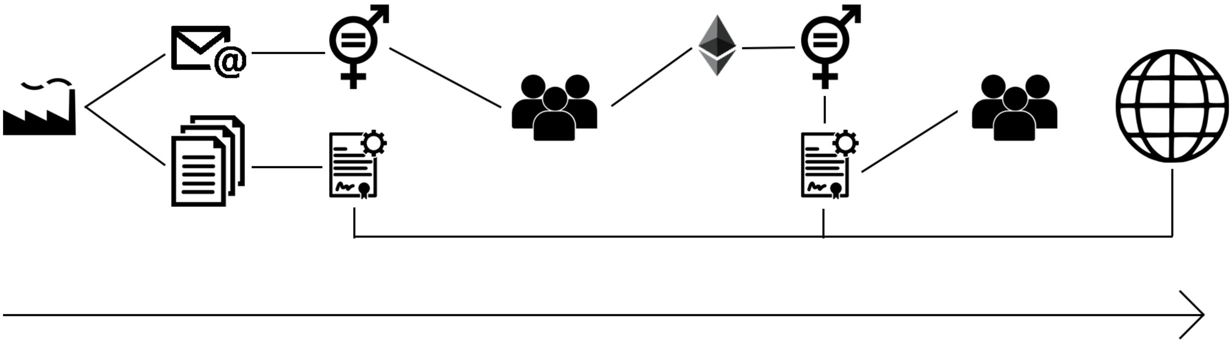

Figure 4: technical flow representation of the data capture and storage process

CANCEL THAT?? (below):

Survey for Employer

The survey handed to the employer, that applies to our gender-equality index, is based on already existing frameworkds such as the Bloomberg gender-equality index or the gender-equality index of Equileap.

-based on existing frameworks such as Bloomberg, Equileap
capturing data with app or web -delivers e-mail addresses from employees
–addresses stored on private database
-survey stored on IPFS
-explain interaction with web/App

We now dive deeper into details of the implementation of the employee survey.

Survey for Employees

The survey for the employees presents the employee with gender specific questions and also with questions applicable for both men and women. These questions guarantee further insight into the microclimate of the employees that cannot be obtained by just consulting the company management.

We earlier explained that we hand our survey to just a randomly chosen percentage of all employees. This is done to increase efficiency of the gender-equality index while cutting costs of applying to that index. The random sample provides us with a hopefully real view of the microclimate of employees. Unfortunately this random sample could lead to a completely wrong picture of the company, we therefore have the option to increase this sample, when there is too much variation from the employer provided data.

capturing data with app or web

-explain,that process automated in the background

-what does the employee has to do

-what is done behind the scenes

storing data: blockchain, IPFS, (some on server)

-how is privacy of data secured

-explain process, what does this mean for different data

–sensibele data – server

–insensible data – ipfs

–non-fakeable – blockchain

-explain cost aspect of storing stuff on blockchain or on IPFS (storing on blockchain costly)

how can anonymity/privacy be ensured

-because data on blockchain, IPFS, it is visible for everybody, but boss should not be able to track the results of employees (how do we solve that)

state technical problems

2.2 Data Verification

why 2 surveys?:

-gives company less chances to actualy deliver wrong information

-how is the verification done

2.3 Display Solution/Evaluation

-introduction about this section -evaluate data and storeed on IPFS:

-link evaluated data from IPFS to an App/Web where it is openly accessible

Multidimensional Approach

———insert picture of spiderweb here

-explain display solution with spiderweb

-link of emplyer, employee survey and overlapped survey

-what can be learned with that

-data connected with respective data on IPFS, blockchain

-open accessible for everybody

-give an example for an indicator an how it could be represented on spider chart (e.g. difference between salaries of men an women).

-explain advantages over existing indices

–better and clearer view, straightforward

–multidimensionality gives a broader picture than just a number

-explain challenges and limitations of the spiderweb

–which indicators can be displayed, which not

2.4 OTHER THINGS TO ADD HERE

todo

2.5 Further Points Worth to Consider

-write about problems that we think that are important, but we didn't have time to consider, or which were to complicated to solve in such a short time
-example: which indicators are best to display in Radar-Chart-Diagram, when is a company considered as a Gender-Equal-Company. How should this be measured, how should different companies best be compared...

3 Conclusion

Strengths

-Our solution provides a reliable, transparent and automatic way of creating a gender-equality index.
-Our approach allows everybody to classify companies according to the latest law regulatory for gender-equality in companies.
-multidimensional
-public
-cheap

Weaknesses

- todo - todo

Open Challenges

-prototype not fully automated
-communication interface app only a dummy
-E-voting system should be improved

Disruptional Potential

-transparency
-further enhance gender equality

4 Sources and Literature

Sources In order to develop the app we used:

- https://developer.android.com
- https://stackoverflow.com

Dateiname - Author - Jahr - Link/Buch