AUDABLOK: Engaging Citizens in Open Data Refinement through Blockchain [†]

Mikel Emaldi, Koldo Zabaleta and Diego López-de-Ipiña *

DeustoTech, University of Deusto, Av. Universidades 24, 48007 Bilbao, Spain; m.emaldi@deusto.es (M.E.); koldo.zabaleta@deusto.es (K.Z.)

* Correspondence: dipina@deusto.es; Tel.: +34-944-139-073

[†] Presented at the 13th International Conference on Ubiquitous Computing and Ambient Intelligence UCAmI 2019, Toledo, Spain, 2–5 December 2019.

Published: 20 November 2019

Abstract: This work describes how open data and human computation can be brought together through blockchain to foster the collaboration of citizens on the continuous enhancement of open data portals. For that, it contributes with a set of enhancements to the widely adopted data management tool Comprehensive Knowledge Archive Network (CKAN), to allow full audit and management of the change requests posed by citizens to datasets in open data portals. User contributions' sustainability in time is tackled by providing rewards to users through AudaCoins, a currency that rewards citizens according to their refinement contributions, thus encouraging their continuous engagement with city co-creation activities.

Keywords: open data; blockchain; smart citizens; user collaboration; co-creation; datasets; crowdsourcing; human computation; citizen engagement

1. Introduction

Human computation, or human-based computation (HBC), is a field that considers the design and analysis of information processing systems in which humans participate as computational agents [1], often as part of their interaction with the real world [2]. On the other hand, internet of people (IoP) [3] is a new internet paradigm where humans and personal devices are not seen merely as end users of applications but become active elements of the internet. It represents the mapping of social individuals and their interactions with smart devices to the internet. It focuses on data collection, modelling, analysis, and ubiquitous intelligence for a wide range of applications of crowd sourced, internet-based personal information. Computation on the edge, not only by making use of blockchain but also making use of people, is a promising endeavour of latest generation computing systems.

Open government data [4,5], i.e., governments' promotion of transparency, accountability, and value creation by making government data available to all, is promising, but the take up has been lower than expected. One of the main issues is usually the long-term sustainability issues of open data repositories. It is common to initiate new repositories, but it is harder to maintain them across time. This impedes a further usage from society and industry of this theoretically very valuable information. Applying crowdsourcing [6] to make open data sustainable without public administration intervention has been previously attempted. Gamification [7,8] and other user-engagement strategies have also been employed to maintain the user motivation towards contributing high quality contents that complement public administration provided information. Anyhow, there are usually some barriers that have prevented a further adoption of this approach:

- Civil servants are reluctant to moderate the contents provided by end-users. Often, they complain that it is harder and less efficient to curate user generated information than to provide this information themselves.

- End users are usually initially motivated, but their contributions are diminished as time passes, since often they do not get feedback from the public administration or they do not perceive any individual benefit out of their altruistic behaviour.

AUDABLOK is our proposed solution to devise a software framework which addresses these challenges, i.e., how to make open government data portals increasable evolvable and sustainable in time. The combination of the human computation approach and the internet of people paradigm are indeed suitable to address these issues. This paper combines end-user intervention and blockchain [9] to realize sustainable open government data portals. It explores how consumers of open data can be turned into prosumers of open data, refining and enhancing contents, by adopting incentivized crowdsourcing, and thus encouraging a more proactive behaviour of users.

2. Related Work

Since the first public administrations started sharing their data as open data, the idea of open government has been disseminated around the world rapidly. As described by the open data barometer [10], Europe leads the region ranking, with widely known initiatives like data.gov.uk (UK) or.opengov.se (Sweden). However, most of countries which have started adopting open government policies have serious deficiencies on exploiting the potential of open data. Many of them are focused only on implementing open data portals, placing little efforts on bringing open data closer to entrepreneurs and citizens through Application Program Interface (APIs). More importantly, many of the available open data portals face ageing issues [11] due to their lack of maintenance and updates.

Trust, immutability, transparency, and traceability are compulsory characteristics for democratic governments. Public administration and services should be grounded in those as they fulfil the minimum requirements to create adequate collaborative environments between different stakeholders. Blockchain is a disruptive technology which is meant to provide a verifiable, immutable, traceable, and decentralized registry of transactions. It was first presented in 2008 to serve as the decentralized public transaction ledger of the cryptocurrency bitcoin [9]. The bitcoin cryptocurrency is the most widely deployed use case of the blockchain technology, proving that blockchain is a disruptive technology for financial sectors but still facing acceptability and governance issues [12]. On top of blockchain, smart contracts, i.e., self-executing scripts that reside on the blockchain, take place. These scripts allow transactions to happen in the blockchain autonomously, avoiding the need of interaction from third parties. Given their features, blockchain, and smart contracts are suitable tools to enhance and complement various public services and automatize public administration procedures [13,14]. One stereotypical case is the one represented by open data portals, as in this work.

The European parliamentary research service's scientific foresight unit advocates blockchain with disruptive effect and impact on lives e.g., public services, such as record keeping without need of a third party, making the processes efficient and affective [15]. In early 2018, PricewaterhouseCoopers (PwC) published a report on potential use and benefits of blockchain and highlighted it as the next innovative technology for making cities smarter [16]. PwC highlighted potential increased productivity, efficiency in processes, and sustainability benefits to cities. These facts are clear indicators that blockchain can revolutionise and change current public administrative processes and procedures. They are aggressively exploring new domains, applications, and workflows where blockchain brings new efficiencies to costly, slow, or unreliable transactions. In the context of smart cities, we believe that further investigations are required to define new models for open governance, partnership, and collaboration by considering the concepts of open governance and active participation of citizens. Hence, this work explores how blockchain can reliably trace user contributions to data and participation in co-creation processes, such as enabling the development of novel business models based on the collaboration and partnership of government and its stakeholders.

A smart oracle [17], in the context of blockchains and smart contracts, is an agent that finds and verifies real-world occurrences and submits this information to a blockchain to be used by smart contracts. In other words, a smart oracle is a data feed—provided by third party service—designed for use in smart contracts on the blockchain. In the context of this work, we have enhanced CKAN [18] tool

with a smart oracle which feeds Ethereum [19] blockchain network, recording the refinement transactions of open data initiated by citizens/end users.

This work aims to improve citizen collaboration through incentivisation and recognition, i.e., trustworthy recording of citizen collaborations. Creating a collaboration ecosystem through the citizens and public administration (PA) is important to develop a sustainable smart city but it could be hard to achieve. Citizens are usually not engaged for collaboration activities due to the lack of agile and efficient channels to do it. Incentivizing citizens for different activities, e.g., recycling, using public transport and so on, is vital for achieving citizen's continuous involvement. However, it is difficult to find really enduring incentives to involve citizens and engage them for enhancing the community. AUDABLOK employs blockchain technology to deal with rewarding and recognition aspects, which we consider key to achieve sustainable engagement from citizens, inspired by previous attempts to seek higher co-creation from citizens through blockchain [20]. AUDABLOK is grounded on our previous work towards more democratized e-services through the involvement of citizens in their commenting and annotation by means of gamification and human computation [21].

3. AUDABLOK Framework

The technical approach, followed by AUDABLOK, is to integrate blockchain, concretely Ethereum [19]—an open source, public, blockchain-based distributed computing platform and operating system featuring smart contract (scripting) functionality—into the open data management software CKAN [20]—an open source data platform which makes data accessible by providing tools to streamline publishing, sharing, finding, and using data. It is arguable that other blockchain networks could have been selected, e.g., Hyperledger [22]. Still, it was important to select a public blockchain network with the same open spirit as the open data movement. Anyhow, the AUDABLOK framework has been designed to make it completely blockchain-network agnostic. Indeed, Figure 1 shows the main blocks that constitute the AUDABLOK framework, namely (a) data lifecycle registry module; (b) blockchain protocol integration module; (c) citizen collaboration module; and (d) data exploitation module. The blockchain protocol integration module acts as a smart oracle, which is blockchain network agnostic and pluggable to other distinct blockchain networks. Notably, AUDABLOK relies on already widely accepted tools to perform: (a) holistic dataset management (CKAN); (b) configuration of a fully decentralized solution (Ethereum); and (c) manage the handling of issues raised as a result of the user-generated contributions to open data (Redmine [23]).

Figure 1. AUDABLOK conceptual architecture.

3.1. Data Life Cycle Registry Module

This module manages the life cycle of data and registers its actions over each of the datasets. CKAN does already cover most of the functionality associated to data life cycle management. However, some extra capabilities which would promote contributions from citizens, like registering all the activities carried out over datasets or the visualization of the data activity registry, are not provided by default. Therefore, this module extends CKAN with the capability to register a new set of lifecycle activities which comprise actions that end users can undertake upon public domain datasets.

Notably, the CKAN data model distinguishes between datasets and resources, being a dataset the meta-package which brings together several resources, i.e., the different files which belong to a given dataset. For instance, a dataset could be composed of several resources which represent the same data in different formats (CSV, JSON, XML, etc.). Another example could be a dataset where the different resources assembling it would be data belonging to the different months of a temporal series. Management of datasets and resources is similar within CKAN. Consequently, therefore they are treated similarly in this work.

Indeed, CKAN currently registers the following activities over a dataset or a resource:

- NEW: reflects that a new dataset/resource has been created.
- CHANGED: reflects that the dataset/resource has been modified.
- DELETED: reflects that the dataset/resource has been deleted.

However, AUDABLOK registers the following extra activities, which are important to record end-user contributions:

- PULL_REQUEST: issued every time a user requests the open data portal moderators to include their contributions over the original resource or a dataset. This request remains in standby until the dataset owner (normally the public administration) accepts the proposed contribution.
- PULL_REQUEST_ACCEPTED: reflects when a request from a user to contribute over a resource of a dataset requested has been accepted.
- PULL_REQUEST_REJECTED: reflects when a request to contribute over a resource of a dataset requested by a user has been rejected.

Figure 2 shows the workflow that must be followed when issuing a request for modification of a data resource managed by CKAN and empowered by AUDABLOK:

1. Metadata generation: firstly, the component retrieves the metadata needed for the block generation. Figure 3 shows the class diagram for package (dataset) and resource classes.
2. JSON document creation: once all the metadata has been compiled, the JSON document to be written into the blockchain is created. In the case of the resources, besides the metadata, the following contents are published:
  - New resource: the checksum of the file is calculated.
  - Resource modification: the checksum of the diff between the original and new files is calculated.
  - Resource deletion: no new material is added, but the file is marked as deleted.
3. Hexadecimal encoding: once the JSON document has been generated, it is encoded into hexadecimal to be sent to the blockchain.
4. Publication request: in this step, a request is made to the system to generate and write the blocks in the blockchain protocol corresponding to the activity performed over the dataset or resource.
5. Transaction storage: in the case that the activity publication over the dataset or resource has been satisfactory, the transaction ID will be published together with the rest of the metadata.
6. Notification: in the case of the dataset or resource publication has not been satisfactory, a notification (e.g., an e-mail) is sent to the dataset author.

In this way, all the activity registered in the data repository is accordingly recorded in the blockchain, thus being audited on behalf of the users of the activity.

Figure 2. AUDABLOK's workflow for management of user-generated contributions.

Figure 3. CKAN's data model for package (dataset) and resource classes.

3.2. CKAN Blockchain Integration Module

One of the most important innovations brought about by AUDABLOK is its capability to trace and audit any activity performed over a public dataset in the blockchain. This module oversees the writing and reading of the different components of the architecture and maps them into queries of the specific blockchain protocol. In this work, we have tested our approach with Ethereum [19]. However, this module can be reimplemented for as many blockchain protocols as needed, e.g. Alastria [24], being easily replaced through CKAN reconfiguration. This is possible by means of a set of actions defined in the CKAN API. If these actions are implemented, any blockchain protocol could be integrated in the platform. The activities performed by such a module are:

1. Integration of user accounts with accounts from blockchain protocol. Each of the users registered in the AUDABLOK platform must have an account for the selected blockchain protocol. A blockchain account is an address at the blockchain in which transactions are registered. The user owning the private key of the account is the only one who can decrypt it and send cryptocurrency from this account to another. During the registration of transactions in the blockchain (particularly for the case of Ethereum), costs that have to be redeemed by the different parties involved in the different processes (datasets publication, algorithms exploitation, etc.) are generated. In AUDABLOK, costs related to the publication of datasets are redeemed by an account belonging to the public administration. On the other hand, citizens must have their own account in order to receive rewards for their participation in the open data portal and consume different services related to datasets.
2. Writing and reading in the blockchain. This module offers the needed operations to write in the blockchain selected, in a transparent manner. It offers the possibility to operate at block, account, or smart-contract level.
3. Reward management support. This module allows citizens who collaborate enriching data to be rewarded, through the desired cryptocurrency.

4. Interaction with the smart contracts. Through AUDABLOK, developers and data scientists are able to promote their algorithms oriented towards enriching and providing added value to the data stored in the repository. These developers can be rewarded through the blockchain protocol selected cryptocurrency (Audacoin in the case of AUDABLOK). Currently, moderators of open data portals grant amounts of cryptocurrency according to the effort and value of the contributions performed by citizens.

Transaction duration on Ethereum depends on the gas price that the user pays when performing a transaction that implies writing in the blockchain, and it is classified on fast (less than 2 min for processing a transaction), standard (less than 5 min), and safe low (less than 30 min). The gas price depends on the value of the ether at the time of performing the transaction. At the time of writing this paper, the gas prices were around 10 GWEI (\$0.037) for fast transactions, 1.5 GWEI (\$0.005) for standard transactions, and 1 GWEI (\$0.004) for safe low transactions. Considering that the process of writing into the blockchain performed at AUDABLOK is done in back-end and no immediate response is needed, standard or even fast safe gas prices could be considered. Audablok saves the checksum of the dataset, not the dataset itself, so transaction time is the same regardless of the size of the resource.

3.3. Citizen Collaboration Module

One of the core features of AUDABLOK is to promote citizen collaboration where they can contribute correcting existing data resources' contents or adding new datasets or resources to the open data portal. For that, AUDABLOK offers a module to promote citizen collaboration which allows citizens to publish their datasets or correct existing datasets resources. The publication of new datasets has been performed following the by-default process defined in CKAN, which has been instrumented to register new dataset transactions in blockchain. Contributions to existing datasets have been defined through an AUDABLOK enabled new process.

This process is grounded on the concepts of fork, pull request, and merge, widely known by those using the Github [25] source code repository. This process is carried in a completely unobtrusive manner through a set of web forms integrated with CKAN, where:

- Fork: describes the process of creating a new branch from an existing repository. The objective of the process is that the user creates its own repository from an existing one, so that they can modify it without interfering with the original repository. In AUDABLOK, this process allows a user to create their own branch from an existing dataset to undertake its modifications and contributions to the composing resources.
- Pull request: is a request to introduce the modified contents by the user in their own branch into the original repository. The user creates a request explaining the modifications and the owners of the original repository decide whether to accept the changes or not. In AUDABLOK, this process allows to integrate the changes performed to resources of the original dataset, if the administrator of the original dataset confirms their acceptance.
- Merge: during this process, the modified resource in the new branch is integrated in the original repository. Two different situations can be given:
  - There is no conflict between the resource in the pull request and the resource in the original repository. Therefore, the integration can be performed in an automatic manner.
  - There are conflicts among both resources, therefore, the admins of the original repository will have to perform the merging manually.

AUDABLOK provides the tools needed to carry out this process in a visual intuitive manner, integrated within the CKAN tool and without forcing end users to use Git-related low level details. Since the volume of change requests which can be submitted in a dynamic highly used open data portal might be big, AUDABLOK is considering integrating with the Redmine [23] issue tracker through its API. Redmine is a tool which would allow a fine-grained tracking of the issues posted through the CKAN enhanced forms provided by AUDABLOK.

3.4. Data Exploitation Module

This module manages everything related with monetizing data and services provided by AUDABLOK. On one hand, this module allows public administrations to reward users for their contributions to the datasets. For instance, by enriching/adding new data, correcting errors, or making improvement suggestions. Besides, users (companies releasing datasets) might also be rewarded when publishing datasets of public interest. This feature has not been implemented in the first version of AUDABLOK but will be considered in future work.

With the aim of incentivising users to enrich the existing datasets or to publish their own datasets, public administrations can establish a set of rewards. That way, they can factor out to the crowd part of the data maintenance and enhancement duties for their open data portals.

On the other hand, this module provides the needed mechanisms for developers and data scientists to promote their smart contracts, which implement the algorithms that can be executed over the data available in the repository. Future work in AUDABLOK will provide the tools to promote and monetize smart contracts. The current implementation of this module only integrates two essential smart contracts to trace dataset pull requests and get rewarded for them when they are accepted, as described in the following section.

4. Implementation and Validation

A fully working initial implementation of the AUDABLOK framework has been completed. Concretely, the following modules have been completed:

1. Data lifecycle registry module—as a CKAN extension, it implements the new operations over datasets and resources proposed by AUDABLOK, namely TRANSACTION_PUBLISHED, PULL_REQUEST, PULL_REQUEST_ACCEPTED, and PULL_REQUEST_REJECTED.
2. Blockchain protocol integration module—an API has been defined which is agnostic to the underlying blockchain protocol. The “data lifecycle registry module” makes use of this module to bridge the CKAN and blockchain network.
3. Citizen collaboration module. The implementation consists of a set of new screens integrated within the CKAN default interface for handling datasets and resources. The purpose of such screens is to map the “pull request” concept from GitHub into AUDABLOK. The current implementation has not been integrated into Redmine but is provided in a basic functional manner within CKAN.

On the other hand, only a first prototype of the data exploitation module has been completed:

4. Data exploitation module. Currently, it incorporates the smart contract responsible for granting AudaCoins to users contributing with information.

4.1. Citizen Collaboration Extensions to CKAN

Figure 4 shows how an end user might request a change of an existing resource by clicking on button “Edit”, which leads a user to Figure 5, a screen where the user can give a title and description to the change requested and directly edit their modifications on the resource editable panel. Once a request for change has been submitted, users and administrator see screenshots to manage pending requests, as the one shown in Figure 6. Finally, Figure 7 shows how a resource administrator might review the changes performed (red are deletions and green are additions). If it agrees with the proposed changes, they can click on “Accept”, granting the user who made the change with the amount of AudaCoins that they have previously agreed on. Alternatively, the administration may reject the change request by clicking on the “Discard” button.

The screenshot shows the CKAN interface for a dataset named 'Questions'. The breadcrumb trail is: Organizations / Societal Challenges / Citizenpedia 2018 / Questions. The page title is 'Questions'. There are 'Download' and 'Edit' buttons. The URL is: <http://localhost:5000/dataset/46844096-c48d-4713-9499-e8f72307b457/resource/61a28e4c-4da5-45f3-9095-5c1cd9096247/download/questions.json>. The dataset abstract states: 'The current package contains the questions and answers created by citizens using the Citizenpedia in the Trento evaluation of the results of H2020 project SIMPATICO that were undertaken...'. The source is 'Citizenpedia 2018'. There is a 'Text' tab and 'Fullscreen' and 'Embed' buttons. A code editor shows a JSON snippet:

```
[
  {
    "_id": {
      "$oid": "59082c7254d661836b9ab7ce"
    },
    "answers": [
      {
        "createdAt": "2017-05-02T06:51:29.706Z",
        "comments": [],
        "stars": [],
        "user": {
          "$oid": "590368bd6bf05468bf476c47"
        },
        "content": "E' possibile perchè l'Accademia nazionale della danza di Roma ha ottenuto il riconoscimento di livello universitario.",
        "tags": [
          {
            "text": "Università"
          }
        ]
      }
    ]
  }
]
```

Figure 4. Dataset description in CKAN.

The screenshot shows the 'Create a new modification proposal' form in CKAN. The breadcrumb trail is: Organizations / Societal Challenges / Citizenpedia 2018 / Questions. The form title is 'Create a new modification proposal'. There are 'Download' and 'Edit' buttons. The 'Title' field contains 'Modification proposal - University tag'. The 'Description' field contains 'Name of the university added to tags'. Below the description is a note: 'You can use Markdown formatting here'. The 'Resource' field contains a JSON snippet:

```
[
  {
    "_id": {
      "$oid": "59082c7254d661836b9ab7ce"
    },
    "answers": [
      {
        "createdAt": "2017-05-02T06:51:29.706Z",
        "comments": [],
        "stars": [],
        "user": {
          "$oid": "590368bd6bf05468bf476c47"
        },
        "content": "E' possibile perchè l'Accademia nazionale della danza di Roma ha ottenuto il riconoscimento di livello universitario.\r\n In base alla Legge 31 dicembre 1999, n.508 quindi ha titolo a rilasciare i diplomi accademici.\r\n Per avere maggiori informazioni ci si può rivolgere alla sede dell'Accademia Nazionale di Danza\r\n Istituto di Alta Cultura: Largo Arigo VII 5 00153 Roma\r\n Tel: 065717621\r\n Fax: 065780994\r\n e-mail: direzione@accademianazionaledanza.it"
        "tags": [
          {
            "text": "Università Trento"
          },
          {
            "text": "Danza"
          }
        ]
      }
    ]
  }
]
```

 At the bottom right is a 'Submit modification' button.

Figure 5. Form for user to submit modification request for dataset administrator.

Figure 6. Screen showing modification revisions pending from administrator.

Figure 7. View seen by administrator to verify changes performed by user, where they have to accept or discard the change request and assign AudaCoins in case of accepting.

4.2. Blockchain Smart Contracts to Redeem Citizens

Within the “data exploitation module”, two smart contracts have been provided. Solidity [26] language has been used to implement these contracts. The first one is the Audablok smart contract (see Figure 8) in which all the changes in the packages (datasets) and resources (create, modify, delete) and branches of them (new change request, accepted proposal, and declined proposal) are saved. Different mappings have been created in which these changes are recorded. An event is emitted when the blockchain is modified in order to have a historical view of the changes done. The second smart contract which has been developed is the AudaCoin contract and its goal is to create and manage the solution’s own coins. People who contribute to create datasets are rewarded with these coins.

```

pragma solidity ^0.5.0;
contract audablok {

    enum Modification { PACKAGE_NEW, PACKAGE_CHANGED, PACKAGE_DELETED, RESOURCE_NEW, RESOURCE_CHANGED, RESOURCE_DELETED, PULL_REQUEST, PULL_REQUEST_ACCEPTED, PULL_REQUEST_REJECTED }

    event PackageEvent(
        string package_ckan_id,
        uint256 timestamp,
        string modification_type,
        address user_address
    );

    event ResourceEvent(
        string resource_ckan_id,
        uint256 timestamp,
        string modification_type,
        address user_address,
        string resource_hash
    );

    event BranchEvent(
        string resource_ckan_id,
        string branch_id,
        uint256 timestamp,
        string modification_type,
        address user_address,
        string resource_hash
    );

    struct BranchUpdate {
        uint256 timestamp;
        Modification modification_type;
        address user_address;
        string resource_hash;
    }

    struct PackageUpdate {
        uint256 timestamp;
        Modification modification_type;
        address user_address;
        string[] resources;
    }

    struct ResourceUpdate {
        uint256 timestamp;
        Modification modification_type;
        address user_address;
        string resource_hash;
    }

    mapping (string=>PackageUpdate) package_updates; // Package updates are saved in this variable
    mapping (string=>ResourceUpdate) resource_updates; //Resource updates are saved in this variable
    mapping (string=>mapping (string => BranchUpdate)) branch_updates; //Branch updates are saved in this variable. If a branch is accepted it is moved to resource_updates
}

```

Figure 8. AUDABLOK smart contract.

4.3. Implementation Details

CKAN is an open source data platform which makes data accessible by providing tools to streamline publishing, sharing, finding, and using data. A plugin for CKAN, corresponding to data lifecycle registry module, has been developed where all the events related to resources and package are tracked, and the modification proposals are managed. The CKAN version used has been 2.8 which makes use of Python 2.7. This plugin calls to function exposed by a REST API, corresponding to the implementation of Blockchain protocol integration module, which is responsible for managing the blockchain transactions. The API has been developed using Python 3 programming language and Flask (1.0.3) web application framework. For the interaction with Ethereum web3.py (4.9.2 version), Python library has been used.

In the implementation of “blockchain protocol integration module”, Ethereum has been used. Ether can be transferred between accounts and used to compensate participant mining nodes for computations performed. Notice that Ganache [27], a personal blockchain for Ethereum development, has been used for the development of the first prototype of AUDABLOK. This tool gives the possibility to manage different user accounts with an initial balance of 100 ETH (ether is the currency in Ethereum) as it is shown in Figure 9. For example, the last account, which has 99,76 ETH and 125 transactions done, is the one that has been assigned to the admin user. Ganache also shows all the transaction done and information related to them.

ADDRESS	BALANCE	TX COUNT	INDEX
0xDe8E37bA8E1e1634e8231e906D113CAff0fA1b66	99.99 ETH	8	2
0xF4c322e41805E36baeDD3c8B9B005086913D21eb	99.99 ETH	5	3
0x5aC8fC91617e700cC3550a53785327c60b05174F	100.00 ETH	0	4
0x0ae1F6b1347Db49d105BE1039305a0707Dd896DA	100.00 ETH	0	5
0x4d4531889Ba2e33714CDB3F5354322080845550	100.00 ETH	0	6
0xd96D69b49Ab764C63e009A71248F3A344c4F7A1c	100.00 ETH	0	7
0x7618290117785D0C0200202bc85248db3F875700	100.00 ETH	0	8
0xC95538871c5f8BEdBfA5314242F273D909004e5f	99.76 ETH	125	9

Figure 9. Ganache development interface.

5. Conclusions and Future Work

This paper has shown how open data and human computation can be aligned by means of adopting a solution based on blockchain in order to support the transparency and accountability principles of open government [4]. AUDABLOK extends the CKAN tool and interoperates with Ethereum to provide a trustful recording of citizen contributions to open data portal enhancements and to grant rewards to citizens for their collaboration activities with public administrations.

Future work should deploy this system through a real open data portal and the actual Ethereum network. It should also further extend the “data exploitation module”, so that a monetization scheme for open data portals can be enabled. The idea is that open data portals should not only be enhanced with the sponsorship by PAs of third party refinements, but private agents should also be willing to host data refinement services in the form of smart contracts which may be consumed and paid for by other users. Data analytics providers will be willing to host their logic within AUDABLOK-enabled open data portals to make them available for customers, and in exchange pay for the hosting, and hence support the continuous refinement of the portal coordinated by a PA.

Future work should also consider extending the AUDABLOK approach to other open government aspects beyond open data, e.g., recycling, usage of public transport, and so on. The very same strategy could be applied to other ubiquitous computing infrastructure with which users interact through blockchain. Hence, citizens would be rewarded when contributing to the city objectives, i.e., using the public transport, buying in local business, interacting with the smart infrastructures of the city, or providing useful ideas to improve the city.

Author Contributions: The main contributors of this work have been: (a) M.E. as creator and designer of the AUDABLOK concept, (b) K.Z. as implementer of the AUDABLOK framework, and (c) D.L.-d.-I. as article editor and creator of the extensibility possibilities of AUDABLOK to other smart city scenarios.

Funding: This research was funded by the Basque Government through HAZITEK program, grant number ZL-2019/00203.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Handbook of Human Computation. Available online: <https://www.springer.com/gp/book/9781461488057> (accessed on 2 October 2019).
2. Bravo, J.; Fuentes, L.; López-de-Ipiña, D.L. Theme issue: “Ubiquitous computing and ambient intelligence”. *Pers. Ubiquitous Comput.* **2011**, *15*, 315–316.
3. Conti, M.; Passarella, A.; Das, S.K. The Internet of People (IoP): A new wave in pervasive mobile computing. *Pervasive Mob. Comput.* **2017**, *41*, 1–27, doi:10.1016/j.pmcj.2017.07.009.
4. OECD. Open Government. Available online: <http://www.oecd.org/gov/open-government.htm> (accessed on 15 July 2019).
5. OECD. Open Government Data. Available online: <https://www.oecd.org/gov/digital-government/open-government-data.htm> (accessed on 15 July 2019).
6. Howe, J. The rise of Crowdsourcing. *WIRED* 2006. Available online: <https://www.wired.com/2006/06/crowds/> (accessed on 15 July 2019).
7. Celino, I.; Cerizza, D.; Contessa, S.; Corubolo, M.; Dell’Aglia, D.; Della Valle, E.; Fumeo, S. Urbanopoly—A Social and Location-Based Game with a Purpose to Crowdfund Your Urban Data. In Proceedings of the 2012 IEEE International Conference on Privacy, Security, Risk and Trust, Amsterdam, The Netherlands, 3–5 September 2012, pp. 910–913.
8. Calegari, G.R.; Fiano, A.; Celino, I. A framework to build games with a purpose for linked data refinement. In *International Semantic Web Conference*; Springer: Berlin/Heidelberg, Germany, 2018; pp. 154–166.
9. Nakamoto, S. Bitcoin: A Peer-to-Peer Electronic Cash System. *Cryptography Mailing List* 2009. Available online: <https://metzdowd.com> (accessed on 15 July 2019).
10. The Open Data Barometer. Available online: https://opendatabarometer.org/?_year=2017&indicator=ODB (accessed on 15 July 2019).
11. European Commission. Ensuring the Economic Sustainability of Open Data Portals: Understanding Impact and Financing. 2018. Available online: https://www.europeandataportal.eu/sites/default/files/s3wp4_sustainability_recommendations_ii.pdf (accessed on 15 July 2019).
12. Ross, E. Nobody Puts Blockchain in a Corner: The Disruptive Role of Blockchain Technology in the Financial Services Industry and Current Regulatory Issues. *Cathol. Univ. J. Law Technol.* **2017**, *25*, 2, doi:10.1108/JFRC-08-2016-0068.
13. Wiig, K.M. Knowledge management in public administration. *J. Knowl. Manag.* **2002**, *6*, 224–239, doi:10.1108/13673270210434331.
14. Zyskind, G.; Nathan, O.; Pentland, S. Decentralizing Privacy: Using Blockchain to Protect Personal Data. In Proceedings of the 2015 IEEE Security and Privacy Workshops, San Jose, CA, USA, 21–22 May 2015; pp. 180–184.
15. Boucher, P.; Nascimento, S.; Kritikos, M. *How Blockchain Technology Could Change Our Lives—In-Depth Analysis*; European Parliament: Brussels, Belgium, 2017.
16. PWC. *Blockchain: The Next Innovation To Make Our Cities Smarter*; PWC: London, UK, 2018.
17. Liu, A. Smart Oracles: Building Business Logic with Smart Contracts. *Ripple Insights* 2014. Available online: <https://www.ripple.com/insights/smart-oracles-building-business-logic-with-smart-contracts/> (accessed on 15 July 2019).
18. CKAN. The World’s Leading Open Source Data Portal Platform. Available online: <https://ckan.org/> (accessed on 15 July 2019).
19. Ethereum.org. Available online: <https://www.ethereum.org/> (accessed on 15 July 2019).
20. Scekic, O.; Nastic, S.; Dustdar, S. Blockchain-Supported Smart City Platform for Social Value Co-Creation and Exchange. *IEEE Internet Comput.* **2019**, *23*, 19–28, doi:10.1109/MIC.2018.2881518.
21. Zabaleta, K.; López-Novoa, U.; Pretel, I.; López-de-Ipiña, D.; Cartelli, V. Designing a Human Computation Framework to Enhance Citizen-Government Interaction. *J. Univers. Comput. Sci.* **2019**, *25*, 32.
22. Hyperledger—Open Source Blockchain Technologies. Available online: <https://www.hyperledger.org/> (accessed on 15 July 2019).
23. Redmine Project Management Software. Available online: <https://www.redmine.org/> (accessed on 15 July 2019).
24. Alastria Blockchain Ecosystem. Available online: <https://alastria.io/en/> (accessed on 15 July 2019).
25. GitHub. Available online: <https://github.com/> (accessed on 15 July 2019).

26. Solidity Programming Language. Available online: <https://solidity.readthedocs.io/en/v0.5.3/> (accessed on 15 July 2019).
27. Ganache One Click Blockchain. Available online: <https://www.trufflesuite.com/ganache> (accessed on 15 July 2019).

© 2019 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<http://creativecommons.org/licenses/by/4.0/>).