

# DESPRO: Decentralized business platform for student non-profit organizations

Matea Vasilj\*, Sven Skender\*, Mia Jurdana\* and Marko Horvat\*

\* University of Zagreb, Faculty of Electrical Engineering and Computing, Department of Applied Computing  
matea.vasilj@fer.hr, marko.horvat3@fer.hr

**Abstract** – Student non-profit organizations are a specific type of small associations that congregate students from diverse backgrounds around a common purpose. These organizations are non-profit driven, autonomous, and typically have largely similar statutes and regulations. In the context of software engineering, information systems that support their business processes have similar software requirements specifications. DESPRO is a web-based platform that fully supports principal use cases of students' non-profit organizations. The platform has three uses: accounting, voting, and fundraising management. DESPRO allows such organizations to begin with their principal operations practically immediately without unnecessary overhead. Unique features that distinguish DESPRO from other similar systems are: 1) lightweight and streamlined deployment process, 2) minimal required remote maintenance, 3) pre-defined functionalities specifically suited for student non-profit organizations, 4) user anonymity customized to specific functional requirements, 5) voting without prior user registration. The platform was developed with open-source technologies Node.js, Vue.js, Vuetify, Bootstrap, Sequelize, Axios, Web3, MariaDB, and smart contracts for Ethereum. It is freely available for non-commercial use by contacting the first author.

*Keywords* - blockchain voting, accounting application, fundraising, decentralized architecture, business application, NGO

## I. INTRODUCTION

Nowadays, there is an increasing awareness of the need for soft skills and networking importance, especially for students while attending college. More than ever, those skills are necessary for creating a successful career which leads students to discovering new and creative ways to improve their skills and at the same time building long term personal, social and business connections. Student associations are oftentimes recognized as a creative and entertaining form for picking up those skills. Students get the chance to organize workshops and other projects while gaining valuable knowledge and experience. Also, through various trips and social gatherings, they meet their colleagues around the world and work on their ideas together. Student associations are also a great opportunity to enter the business world, bookkeeping and fundraising are just some of the areas students are introduced to [1].

Non-profit organizations are the prevailing model for student associations. In such organizations not only soft skills, but also organizational skills are gained, as well as a grasp of management. It all sounds great, but it tends to be

logistically very challenging to bring this to reality and make it sustainable over time. Organizations are legal entities with legal issues and their members as electorate. Each organization must have its representatives. Representatives are chosen periodically by members through an act of anonymous voting. Since it is a non-profit organization, representatives are also handed the responsibility to fundraise with the help of members. Last, but not the least, they have to make sure all the associations financial transactions are booked according to relevant financial legislation.

Accounting, anonymous voting and fundraising can be recognized as main business requirements of any student associations [2] [3] [4]. Therefore, these three functionalities must be implemented as core functionalities of any business platform. In this regard, we have developed a specialized custom-tailored and web-based platform that fully supports principal use cases of students' non-profit organizations.

The DESPRO (DEcentralized buSiness PlatfoRm for students' non-profit Organizations) platform was developed with a motivation to digitize and improve the efficiency of everyday work for students in their associations. At this point, the platform is customized for one specific non-profit student organization, but it was built with the attention of easy integration for any other similarly functioning organization. Software is free for academic and non-profit use and can be obtained by contacting one of the authors.

The remainder of this paper is organized as follows; Section 2 gives an overview of the platform and its system architecture, Sections 3 illustrates the accounting module, Section 4 provides information about blockchain as core technology for voting and Section 5 embodies fundraising tools. Finally, Section 6 concludes the paper and provides insight into next steps.

## II. PLATFORM ARCHITECTURE

The DESPRO platform is distributed with front-end code written in Vue.js, back-end code written in JavaScript for Node.js engine and a series of smart contracts for Ethereum Virtual Machine (Solidity v0.7.0.). Smart contracts come along with migration files which are needed to publish them on the Ethereum blockchain. To successfully build and run client and server code, Node.js v15.5. as well as Node Package Manager – npm v6.14. or newer is required. MariaDB 10.5. series database must be

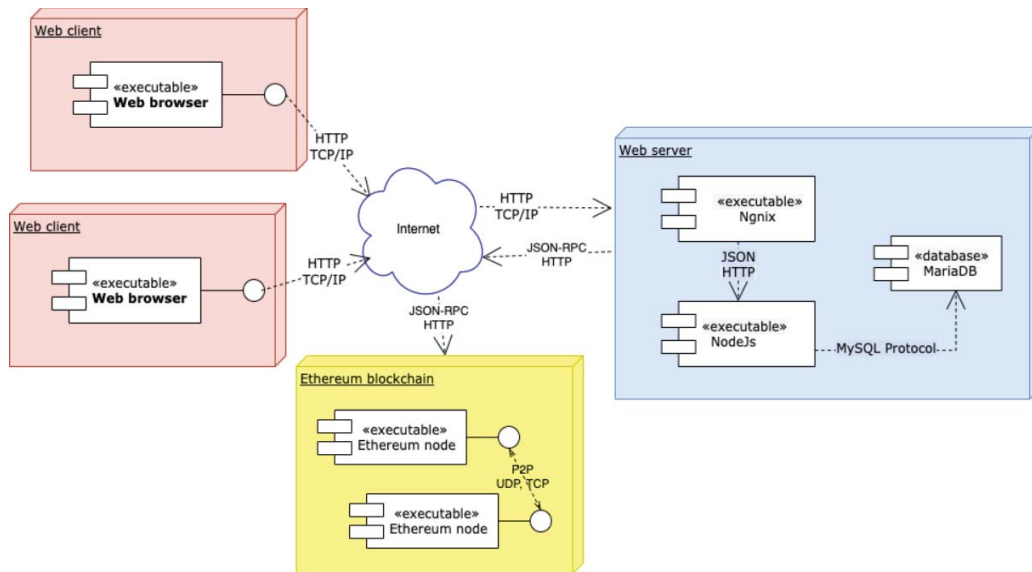


Figure 1. The DESPRO web-based platform's UML component and deployment diagrams.

preinstalled on the server. To serve the content to users, a HTTP web server (per example, Nginx or Apache) must be installed on the server machine, too. Architecture of the platform is illustrated in Figure 1.

Before running the platform, a new user is ought to be inserted in the database followed by a creation of an empty database. The user is expected to have full permissions on the created database. Database administrator is not required to create or import any SQL tables up front, because the backend will automatically take care of that.

Although smart contracts are shipped with the platform source code, after the initial migration to the Ethereum blockchain network, they are no longer treated as a dependency. The contract deployer becomes a so-called contract owner and all further interaction with contracts goes directly through the voting web application and users' wallets via Ethereum JSON-RPC API.

The back-end part of the platform is designed to read configuration data, such as database connection string, listening port, etc., from environment variables. Environment variables can be exported in a shell or saved to an .env file that will be read upon a start-up of the Node.js engine. No changes are required in the code base. The front-end part of the platform is designed on the same principle. URI strings for REST API server and Ethereum nodes, as well as origin addresses of deployed smart contracts are read from the .env file which must be present during the build process. After the build is finished, the content can be migrated and served to clients by one of the available HTTP web servers.

For a quick start and a painless deployment of the platform as a whole, a script for Docker is present in the root folder. The script builds, containerizes and runs each platform module as an independent Docker service. Such services are easily scalable as they can be run on a demand in numerous clusters across multiple server machines. As of now, Docker script runs the following

services: MariaDB database server with a new user and an empty database, Node.js for serving back-end, Vue.js build followed by Nginx as a HTTP server and finally – private Ethereum blockchain network node (Geth client v1.9.25.) as shown in the Figure 1. For testing purposes, it is recommended to use a private Ethereum blockchain as it behaves the same way as Ethereum mainnet, difference being that transactions on the test network are free of charge, and on the mainnet each payment requires a real Ether (ETH) which is worth actual money.

### III. ACCOUNTING MODULE

From the main screen, the user can select the left option “Knjigovodstvo” (Accounting) to access the platform's accounting module. This module was intentionally designed to support the simplest legislative version of the bookkeeping processes, because most of student organizations need to follow only this type of bookkeeping. Accounting personnel in this type of student organizations encounter many obstacles that lead to creation of such solution. This particular student organization tried to use some of the available software solutions, but none of them were enough user friendly nor did they match its use case. In order to finish all required bookkeeping, user has to enter only four types of data and the system will automatically generate all the necessary reports, so no previous experience of specialized knowledge of accounting is necessary.

The required inputs are as follows; first it is necessary to register financial balance at the start of each year, both in bank account and in the treasury. As soon as user types those in, the current balance gets updated and the user can start to record incoming and outgoing accounts. Since most of the accounts were issued by a few companies, it is mandatory to record company information to prevent non consistent inputs and decrease data redundancy. After all the necessary information to generate valid reports is available, it is possible to generate these reports. In accordance with appropriate Croatian financial legislation,

all the legal entities that do this kind of simple bookkeeping, have to file five specific reports and have them ready at any time. The reports must be in an Excel spreadsheet format. In order to generate any of these reports, the user only has to select the time period and type of the report, and the spreadsheet will be automatically downloaded to the user's computer. There are only a few things to keep in mind; account balance mentioned before will be updated on each account entry, but using advanced business rules, will also prevent you from making mistakes while doing your bookkeeping such as recording duplicate entries, spending more than you own etc. There are many other business rules implemented, so the system will let the user know if any misstep has occurred using user friendly messages and suggestions on how to correct them. It will also prevent users from deleting or making illegal changes on data itself.

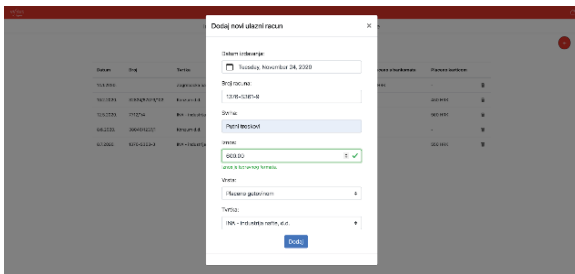


Figure 2. Inserting incoming account in the Accounting module.

In Figure 2 we can see how easy it is to record an incoming account. The user only has to manually insert account number, purpose and amount and after that he has to pick the date, source and the issuing company. All other inputs use the same logic.

#### IV. BLOCKCHAIN MODULE

After navigating to the voting page, a voting panel is presented to the user. Voting panel consists of ballots. Any member can submit a new ballot with a proposal at any time. On the other hand, ballots with candidates can be created only by the organization's chairman. Once submitted, both are stored on a blockchain and cannot be modified afterwards [5] [6]. Each ballot is active only for a specific period of time defined by the interval between two block numbers. The start and end blocks, which define the time limit, are stored on the blockchain. So, after the end period has expired, voting becomes unavailable and the outcome is automatically calculated by an algorithm defined in a voting smart contract. For a proposal to succeed, due to the legal statute of the student association, at least fifty percent of members are required to give a vote and fifty percent of those voters must vote in favour.

To start working with the blockchain module, a user must connect a wallet to the voting web application. This can be any wallet which is available for use in a web browser. This is supported to use wallets in a form of software wallet like Metamask or Brave's wallet as shown in Figure 3. Hardware wallets, such as Ledger or Trezor, are also supported. After the process is done, the voting module automatically reads the public address of the connected user and checks whether the user has a valid

membership and voting rights. Membership is represented by holding an organization's ERC777 vote token. One is allowed to vote if he is a legitimate organization's member [7] [8] [9] [10].

Next step for the user is to click on an active ballot with a proposal or a candidate and vote in favour or against, or as sustained. This transaction is now signed from the user's wallet, recorded on the blockchain and the ballot is no longer available for the user to vote nor change his opinion.

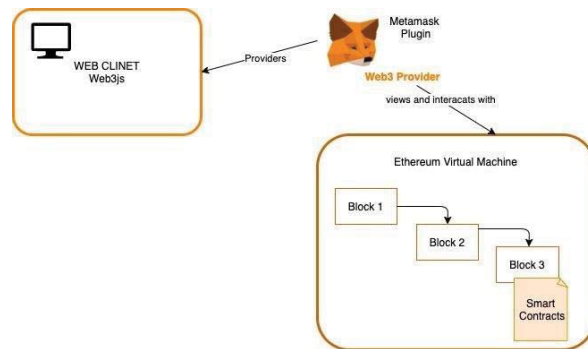


Figure 3. Interaction with voting through the Metamask plugin in the Blockchain module.

Voting on a blockchain is quite appealing to members. They can vote from any place in the world yet remain assured that obligatory requirements for a voting system are satisfied. The set of requirements for the blockchain module has three important items:

- immutability: no one can tamper with the vote after it was made,
- authorization: only members can vote, and nobody can vote more than once,
- privacy: relation between a voter and his vote is kept in secret [11] [12] [13] [14].

#### V. FUNDRAISING MODULE

Last but not the least, the third set of functionalities in the DESPRO platform is implemented in the fundraising module. As discussed earlier, most of the student organizations are non-profits, which makes fundraising one of the essentials to keep the organization on track. Fundraising can be split in two categories, material and financial. Timewise, it is possible to get funded by the project or by time period. Project funding is the most often used model, so the system is especially adapted to it. The process starts by creating a new workshop, defining the appropriate timetable, topic and similar details and most importantly, quotation for potential project funders as seen in Figure 4.

The system will then guide the user through a process similar to lead generation often used in business applications [15]. It consists of entering contacts of potential funders and assigning tasks to keep user on track with each potential funder. All tasks can be found on a workshops timeline displayed in the graphical user interface.



Figure 4. Creation of the new workshop with a quotation for potential project funders in the Fundraising module.

Using the Fundraising module, the user can close a specific opportunity as won or lost. In case of a win, the agreed funds are recorded as well, followed by an update of the agreed offering. For all other sponsorships, the process is similar. For simpler time organization there is an option to export a calendar entry to make it easier for a user to personally track tasks and workshops. There is also one additional benefit: new organizers can always look back to previously workshops to find best management practices. There is one additional functionality and it is the workshop partner contacts view. The user can look at the list of all contacts that were previously entered and update or delete them later.

## VI. CONCLUSION

To sum up, DESPRO is a platform that fully supports core business functionalities of students' non-profit organizations. We believe that the development of the platform followed principals of development of primary and secondary experience of adaptive information systems supporting knowledge transfer [16] [17] [18].

Looking ahead, there are a great number of possibilities for an upgrade. Simple as it may be to insert new accounts in the system, it would be even better if one could just take a photo of the receipt and the system would automatically digitize the data. Bookkeeping and fundraising could go hand in hand if the system could generate an outgoing account from the fundraising module and adjust bookkeeping accordingly.

Having in mind mentioned organizations mostly deal with money and more and more payments are expected to arrive in a form of cryptocurrency, it would be attractive to have some kind of an on-chain vault. Such a vault would store incoming payments converted to one of stable coins. No one would be able to withdraw funds on his own, instead, all members should be obligated to vote on

how to spend those assets. In addition, funds could be supplied as a collateral to another DeFi project [19] [20] [21]. In this way, not only an interest would be earned in that scenario but also additional assets could be borrowed in case that the organization has provided enough collateral.

Regarding the fundraising module, the next goal would be to set up reporting that would show likelihood for each potential contact of becoming a sponsor. Calculations would be performed based on tasks tracked for that specific contact and previous agreements. Also, we aim to unify the process for every workshop in the same funding category to make it easier for organizers and for the companies as well. It would consist of specific consecutive steps that have to be completed in each stage of negotiations. We believe these improvements would lead to more successful student organization funding with less resources used.

## REFERENCES

- [1] R. Tian, "Strategies for the Organic Integration of the Construction of Student Associations and Professional Practice in Applied Technology-Based Universities", International Conference on Economics, Education and Social Research, China, 2020.
- [2] L. Francisco and M. C. Alves, "Accounting information and performance measurement in a nonprofit organization", *Performance Measurement and Management Control: Global Issues, Studies in Managerial and Financial Accounting*, vol. 25, pp. 465–487, 2012.
- [3] L. Najev Čaćija, "Fundraising in the context of nonprofit strategic marketing: Toward a conceptual model", in *Management : Journal of Contemporary Management Issues*, vol. 18(1), pp. 59–78, 2013.
- [4] M. Swan, "Anticipating the Economic Benefits of Blockchain", in *Technology Innovation Management Review*, vol. 7, pp. 6–13, 2017.
- [5] Ethereum Whitepaper, <https://ethereum.org/en/whitepaper>, accessed 7 February 2021.
- [6] S. Nakamoto, *Bitcoin: A Peer-to-Peer Electronic Cash System*, Manubot, 2008.
- [7] R. Leshner and G. Hayes, *Compound: The Money Market Protocol*, available at White Paper, 2019.
- [8] W. Reijers, F. O'Brolchain and P. Haynes, *Governance in Blockchain Technologies and Social Contract Theories*, Ledger, vol. 1, 2016.
- [9] The Maker Protocol: MakerDAO's Multi-Collateral Dai (MCD) System, available at White Paper.
- [10] I. G. A. Pernice, S. A. Henningsen, R. Proskalovich, M. Florian, H. Elendner, and B. Scheuermann, "Monetary stabilization in cryptocurrencies - design approaches and open questions.", *Crypto Valley Conference on Blockchain Technology (CVCBT)*, pp. 47–59, 2019.
- [11] S. Barański, J. Szymański, A. Sobiecki, D. Giland and H. Mora, "Practical I-Voting on Stellar Blockchain", in *Applied Sciences*, vol. 10(7606), pp. 1–9, 2020.
- [12] N. Kshetri and J. Voas, "Blockchain-enabled e-voting", in *IEEE Software*, vol. 35, pp. 95–99, 2018.
- [13] D. Khoury, E. F. Kfoury, A. Kassem and H. Harb, "Decentralized voting platform based on ethereum blockchain", in *IEEE International Multidisciplinary Conference on Engineering Technology (IMCET)*, 2018.
- [14] E. Yavuz, A. K. Koç, U. C. Çabuk, and G. Dalkılıç, "Towards secure e-voting using Ethereum blockchain", *6th International Symposium on Digital Forensic and Security (ISDFS)*, 2018.
- [15] Qualify or convert leads (Sales and Sales Hub), <https://docs.microsoft.com/en-us/dynamics365/sales-enterprise/qualify-lead-convert-opportunity-sales>, accessed 7 February 2021.

- [16] S. Lugović, I. Dunder and M. Horvat, "Primary and secondary experience in developing adaptive information systems supporting knowledge transfer", 40th International Convention on Information and Communication Technology, Electronics and Microelectronics (MIPRO), pp. 1207–1210, 2017.
- [17] S. Lugović, I. Dunder and M. Horvat, "Secondary Experience of an Information System Enabling Scientific Communication," Proceedings of the 1st International Communication Management Forum (CMF2015), pp. 562–587, 2015.
- [18] S. Lugović, I. Dunder and M. Horvat, "Primary and Secondary Experience as a Foundations of Adaptive Information Systems", in Proceedings of the ISIS Summit Vienna, 2015.
- [19] H. Amler, L. Eekey, S. Faust, M. Kaiser, P. Sandner and B. Schlosser, "DeFi-ning DeFi: Challenges and Pathway", at arXiv:2101.05589, vol. 1, 2021.
- [20] C. R. Harvey, A. Ramachandran and J. Santoro, "DeFi and Future of Finance", at Social Science Research Network (SSRN), pp. 23–35, 2020.
- [21] D. A. Zetzsche, D. W. Arner, and R. P. Buckley, "Decentralized finance", Journal of Financial Regulation, vol. 6(2), pp. 172–203, 2020.