Peer to Peer Systems and Blockchains
Final Project
Academic Year 2018/2019

Development of a Dapp for Smart Auctions

1 Goal of the Project

The final project consists in an extension of the final term assigned in this academic year whose specification is available at the link `https://elearning.di.unipi.it/course/view.php?id=155`, on the Moodle page of the course. The final term required to implement and test (virtually via Remix) the smart contracts for two smart auctions systems. This project requires:

- to write the frontend of the smart auctions.
- to deploy and test the implemented system on a real blockchain.

2 Smart Auction Dapp: implementation

The student is requested to develop a Dapp which must include two components: the blockchain-based backend and the web-based frontend. The backend is implemented through the set of smart contracts implemented in the final term, while the frontend must be implemented from scratch. The student may use web3 Javascript, Java or Python as base language.

The Dapp provides different interfaces, one for the bidders, one for for the auctioneer and another for the seller. It is required to implement a function *Create Auction* which is used by the auctioneer to create a new auction. This function creates an event which is notified to all the potential auctioneer who can join the auction. Later, another event notifies the bidders of the auction start. It is required to implement all the functionalities needed for the auction choosen in the final term.

The frontend asynchronously receives, through a set of callbacks, a set of events emitted by the smart contracts. Students are requested to define these set of callbacks, which must include at least those triggered by the main events happening in the aution.

Do note that, since web applications are not the main topic of the course, the user interface of the frontend of the Dapp will not be taken into account for grading. As long as it is working, a basic GUI is all that is required. Still blinding colours, fixed 1024x720 windows and other examples of unusable UIs will not be appreciated.

3 Smart auction deploy

The Dapp should be implemented on a real blockchain, in particular, it must be deployed on the Ropsten Ethereum testnet. Students can obtain the needed test ether from a faucet as explained during the course. All deployed contracts are required to have a suicide operation to relieve the testnet after the experiments. Furthermore, students are advised to perform their trials on a local testnet started by them and only test the final project on Ropsten to ease the burden on the global testnet.

4 Project Submission Rules

The project must be developed individually. Students should submit:

- the frontend code (developed in any chosen language), as well as the instructions to compile, deploy and run the Dapp.
- a brief report including the main project choices, a list of operations to test that the Dapp is working as intended.

The report and the code must be submitted both electronically, through the Moodle, and at the reception desk of the Department of Computer Science. The project will be discussed a week after its submission. The discussion consists in the presentation of a short demo and a general discussion of the mid term, of the final term and of the choices made in the implementation of the project and of the developed code.

The oral exam is needed only if the student has not passed the Mid or the Final term and regards the only the part not passed.

Do not hesitate to contact us by e-mail (laura.ricci@unipi.it) or during the question time, Thursday 15.00 PM-18.00 PM.