# P2P Systems and Blockchains
# Final Term
# Academic Year 2021/2022
# TRY: a nfT lotteRY

## 1 Project Description

The goal of the project is to implement **TRY**, a lottery offering users collectibles as winning prizes. The rules of the lottery are inspired by *Powerball*, a popular type of lottery played in the United States. The lottery logic must be implemented with a *Solidity* smart contract on the *Ethereum* blockchain. Before operating the lottery, the lottery manager buys a batch of collectibles, like those presented in Fig. 1 and mints a Non Fungible Token (NFT) for each of them. Each lottery's winner may receive as a prize one of these collectibles.

**Figure 1** Lottery's Collectibles

## 2 TRY: implementation

In Powerball, a user buys a set of tickets and picks six numbers per ticket. The first five numbers are standard numbers from 1- 69, and the sixth number is a special Powerball number from 1 - 26 that offers extra rewards. Each ticket has a fixed price.

Like Powerball, TRY is a recurring lottery with multi ticket purchases and multiple payouts. The lottery is executed in different rounds, and each round has a fixed duration of M blocks. A new round may only be opened by the lottery operator. Opening a new round is allowed the first time, when the contract has been deployed, or when a previous round is finished. A round is *finished* when the winning numbers have been drawn and the prizes have been assigned. A round is *active* if the M blocks defining the duration of a lottery round have not yet been confirmed. The user can buy a ticket only during an active round, if they buy a ticket when no round is active, the contract returns an error message. The operator can close the lottery at any moment. If there is an active round, the users receive back the total amount corresponding to the prices of all the tickets bought in that round. When a round is finished, the total balance of the contract (corresponding to the revenue of the tickets sold in that round) is transferred to an address provided by the operator.

For each round, a drawing is held, and a winning ticket consisting of five standard numbers and a Powerball number is picked. Prizes are paid out based on the number of winning numbers matched on the user's ticket.

Before starting the first round of the lottery, the lottery's operator creates a Non Fungible Token for each collectible, and defines the value rank of that collectible. The operator may mint further NFTs during the lottery, if needed. To simplify the implementation, the NFT may contain a description of the collectible's image (or the NFT may store the image itself or the URL of the image, if an image repository is available).

The collectibles are divided into eight classes (not eleven) as explained in Fig. 2, each class corresponding to the matches of numbers in a draw. For example,

as shown in Fig. 2, a Kitty of Class 3 can be won if a player's ticket matches 4 numbers and the powerball number in the draw. The assignment of the collectibles to the classes is random, and the choice of a particular collectible in a class as prize is random as well.

The operator then draws randomly 5 numbers and the powerball number. To generate a random number, it is mandatory to use a source of randomness coming from the blockchain itself, and decided a priori before the lottery starts so that the operator can not influence the result. For instance, for each round R, the hash of the block of height at least X+K, where X is the height of the block corresponding to the end of R and K is a parameter. Each user may receive a prize from the lottery's operator for each one of its tickets.

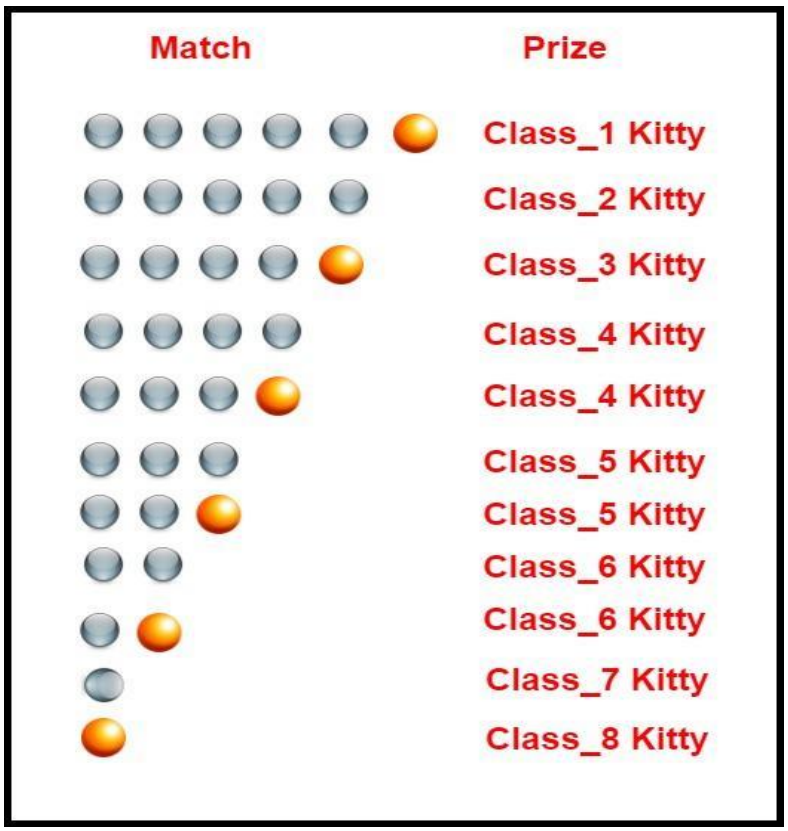


**Figure 2** Lottery's Prizes

Figure 2 shows the correspondence between the number of matches (the golden ball represents a Powerball match, the gray all other matches) and the prize assigned to the user.

# 3 Requirements

The student must:

- integrate in the project an NFT contract implemented according to the ERC721 standard, that is the de-facto standard for non fungible tokens. It is possible to use an existing implementation of the contract, but it is required to describe in the report the functionalities of the ERC721 contract used by the TRY application, together with their description. It is also possible to add other functionalities/data to the contract, if their introduction is properly justified.
- write the lottery smart contract, which will implement at least the following functions:
    - *startNewRound:* checks if the previous round is finished, and, if that's the case, starts a new round.
    - *buy*: allows users to buy a ticket. The numbers picked by a user in that ticket are passed as input of the function. The function checks if there is a round active, otherwise the function returns an error code.
    - *drawNumbers*: used by the lottery operator to draw numbers of the current lottery round
    - *givePrizes*: used by lottery operator to distribute the prizes of the current lottery round
    - *mint:* used to mint new collectibles
    - *closeLottery:* deactivate the lottery contract

  the smart contracts must be developed in *Solidity*, and compiled/deployed on *Remix*

- log important events generated by contracts notifying the outcome of a set of operations.
- provide a list of operations (contracts deployment, transactions sent, etc...), temporally ordered, enabling to run a meaningful simulation of the system.
- provide an estimation of the gas consumed by at least one of the non trivial functionalities implemented by the system.
- analyze the security of the implemented smart contracts, in particular discuss if the random number generation on the smart contract may be a vulnerability for miner attacks.
- write a brief report discussing the previous points

The assignment must be done individually and its deadline is **6 June 2022**. The assignment can be submitted even if the mid term has not been submitted or has not been passed. The assignment must be submitted through Moodle. Its evaluation will be notified through Moodle as well. The assignment is not mandatory, if it is not presented, the student will be required to pass the oral exam on the second part of the course.