



3327 Internship

Assignment

Dear Leon,

Thank you once again for sharing your interest in joining the 3327 crew during the summer.

We created this task to test your motivation and to make sure that you are a fast learner and self-driven individual capable of comprehending the problems and offering potential solutions on your own.

Hence, below you will find more information about the challenge of creating a decentralized betting platform.

Once you finish with the task, please share the GitHub repository with your solution and deploy smart contracts on the testnet of your choice.

Betting.sol

Our task is to create a decentralized betting platform for sports events powered by smart contract(s) written in **Solidity**.

The MVP version of our Betting platform should support only three main betting types:

1. Tie
2. Team A is going to win
3. Team B is going to win

Also, betting is supported only in ether (or any other native coin depending on chain) for now.

Create a *placeBet* function that takes *matchId*, *bettingType* and *oddForWinning* as arguments. Bet can't be placed after the betting deadline. We can assume that the betting deadline is the same for all matches and can be set inside a *constructor*. If the user provides 0 (zero) ether, the function call should revert. Function should emit proper events.

Create a *payWinningBets* function that takes *matchId* and *winningType* as arguments. This function should be callable only by the Betting Administrator. It should transfer the winning amount to all Users which have won the bet. Winning bet equals initial amount times odd minus 5% fee. (This basically means that 5% of *initialAmount * odd* remains in the contract as fee and 95% of *initialAmount * odd* goes to the User). Function should emit proper events.

Deploy to the testnet of your choice.

Betting API

Create a simple **Node.js** API which provides information about sport events our Betting platform supports. Each sport event must have a unique matchId and proper odds. It should also have additional info such as Team Names, Match Date, etc.

Example:

```
{  
  "matchId": "1",  
  "competition": "UEFA Champions League",  
  "teamA": "Bayern",  
  "teamB": "PSG",  
  "Tie": "3",  
  "Team_A_Win": "2",  
  "Team_B_Win": "3",  
  "gameDay": "August 23, 2020"  
}
```

Bonus: Oracles

Since odds can change during time, discuss how we can improve our Betting platform using blockchain oracles.

Good luck!