It is a multi-player game dapp. A demo can be found here: https://dice.game/play

Game flow and logic details

Every 30 seconds three random dices are chosen and shown in the results, and results are saved to the db. Depending on the results winners are shown based on the winning logic (winning amount logic is displayed below each option in the game board)

One game time 30 seconds - betting option will be closed before 5 seconds and shows a stop betting image, and results shown for 5 seconds and then repeat the whole process.

User can select the amount of DICE chips (1, 5, 10, 50, 100, 500 and 1k) they wish to place on the game board to play the game >> they can place it on the game board on the option they like >> they click on the BET button and user bets are saved to the DB with their wallet >> waits for results to come >> once the result is shown, if they win, they get DICE chips according to the amount they won.

Users can continue playing, it's multiplayer and the game keeps playing whether or not users are playing, and results are shown in the right side, check the reference game also if you'd like to know how it's already done >> https://dice.one/sicbo

The board will continue to play and show new results, whether there are players or not, it continues and results will be displayed in the right side panel; current users' bets will be displayed in the left side panel.

How does it work (Website connection requirements)

It's a simple dice game. Demo of the interface and how bets and winning works can be found in the demo above. Users will need DICE tokens/chips to play the game

Game Play Step 1: User connects wallet (metamask or trust wallet - BSC)

Game Play Step 2: User swaps some BNB to get DICE tokens (smart contract will have a swap option and liquidity option) - only one swap option BNB to DICE and DICE to BNB (User clicks on SWAP in the menu and they will be taken to the swap page)

Game Play Step 3: User selects the amount of DICE tokens they wish to place on the game >> they places it on the game board (on the number or option they like) >> they click on the BET button and waits for results to come >> once the result is shown, if they win, they get DICE tokens according to the amount they won

They can continue playing or swap again (to get DICE or BNB) as they wish.

The board will continue to play and shows new results every minute, whether they are players or not, it continues and results will be displayed in the right side panel; and users' bets will be displayed in the left side panel.

Smart Contract Requirements

Token name: DICE - BSC, bep20 token

Initial mint/supply 10,000,000 DICE (10 million DICE). 1,000,000 DICE from this will be placed in the game contract to provide enough DICE tokens to win, for users to play the remaining 9,000,000 DICE will be sent to the developers.

Maximum total supply is 1 billion.

10 new DICE tokens will be created per every block till the total supply reaches to 100 million DICE

50% of the total tokens created per block (in a 24 hours period - initial mint is not included) will go to the user addresses who played in that 24 hours period according to the number of times they played. Logic for user reward is (Total tokens created in 24 hours period) * (Total Games Played by a user/Total games Played in the 24 hours period). If 200 tokens are created in a 24 hours period, and 50 total games were played, one user who played 2 games will get 100 (50% of total) * 2/50 = 4 DICE tokens

30% of the total tokens created in a 24 hours period will go to the Liquidity providers based on the % of liquidity provided. Reward logic same as above Total Tokens (30% of total) * user-liquidity/total-liquidity.

10% of the total tokens created per block in a 24 hours period will go to the developers.

10% of the total tokens created per block in a 24 hours period will go to the game contract.

Liquidity pool function

This allows users to swap BNB and DICE tokens. Liquidity pool users will get DICE for their stake every 24 hours. (developers will add an initial liquidity pool for the swap to work)

SWAP function

To allow users to swap between BNB and DICE. (developers will add an initial liquidity pool for the swap to work)

BURN function

To allow the developers to burn a certain amount of DICE tokens when needed.

BNB Rewards function for liquidity pool providers

In addition to the daily DICE rewards for the Liquidity Pool providers, they will also get BNB every 24 hours.

Here's how it is calculated.

If the total number of Dice tokens in the game contract is greater than the total number of DICE tokens of the previous day, 30% of the additional DICE tokens will be swapped to BNB and this BNB will be distributed among the liquidity providers accordingly

For example, if the previous day had a total of 1000 DICE tokens in the contract, and the current day has 2000 DICE tokens (the newly minted tokens added to the game contract will not be counted in this calculation), the difference is 1000 DICE tokens; so 30% of this is 300 DICE tokens; this will be swapped to BNB and sent to the Liquidity Pool providers proportional to their stakes.

This BNB reward happens only if the total number of DICE in the current day is bigger than the previous day.

Other functions for the game/smart contract

Maximum Bet amount (maxwager variable) will be 10% of the total DICE tokens in the game contract (to make sure there's enough for new games to be played. So the maximum bet amount will be like 10% of the total number in the game contract - winning amount needed for bets that are already placed)

There is a limit for bet placement, users won't be able to place unlimited amount; they will only be able to place amounts for which the game contract has sufficient DICE to spare if they win. So if the game contract has 1000 DICE; 10% of that is 100 Dice; So a user who places a bet in winning chance 1:2 (if he wins he gets double of what he placed in the bet) can only place a maximum of 50 DICE. Since multiple users are there, each time when a user makes a bet; the possible winning amount needed for that user bet will be deducted from the maxwager variable; so the game never fails.

VPS Server details cpanel: https://dice.game/cpanel
user: dicegame
pwd: @&2Freelance

The game is in the folder /play (as in dice.game/play)

Source code file is source_game.zip

Other notes:

1. Past results section will always be full with previous results (results are saved to db and shown in the right side results panel - just like in dice.one/sicbo). The latest result will be shown at the top. The result before that is shown below that and so on.. so users can scroll down and see all the previous results. Results are the same for all users. The game doesn't just start when someone opens the website; the game keeps running (on the server) and results are stored whether anybody is playing or not. When a user (or any users from anywhere in the world) visits the website, they see the same game, not a different game. Right now the game stops when a user moves away from the window, that's not how it needs to be.

2. A variable named **maxwager** (maximum bet amount) can be added to the db - to limit people from placing indefinite number of bets. You can see maxwager in dice.one/sicbo in the game board

3. Can save results and user bets to DB; user bets can be saved to db with the chips they bet and IP address as name since it's multiplayer (id, time, username as IP and user bet chips, maybe the associated game ID or number too)

4. The database details we currently have in mind >> Results: result-id, result-timedate,result-dices ~ Users: user-id, user-wallet ~ Bets: bet-id, bet-timedate, bet-user, bet-option along with these we will also have to add a game-id as well.