**3d rotating Phaser 3 Dice Class**

**Quick start guides:**

To run the dice program, please follow the below steps

1. Unzip the final files.
2. If you open index.html directly in the browser it will not work.
3. Upload to a local server or if using node.js, start http-server by using `http-server` from NodeJS command prompt in the same folder where your game resides.
4. Locate to the path in the browser: localhost/”game folder path”
5. Enjoy the game and start editing.

**Basics**

Quad: A quad is a container which has four points to draw two quadrilateral triangles with four points. The four points are being used to set the image’s four corners.

Random Number: The Dice takes a random number to stop at when user rotates it or rolls it.

Customizable Texture: One can create any kind of dice with 6 faces with this code.

**How to customize dice texture?**

One can create the image sprite of six sides of a dice and put the images in the application. Like below,

this.load.spritesheet('d1','d1.png',{frameWidth:80,frameHeight:80});

….

var quads1 = 'd1';

….

new Dice(quads1, 145, 100, model.verts);

Dice size: One can easily set the dice size from **`var diceSize = 26;`**

**Code Help**

Introducing a Phaser 3 dice class written in Object Oriented JavaScript. This includes intermediate level of a Dice Class.

The Phaser 3d rotated dice class (Object Oriented Development), which allows you to customize its texture/sides and allows a number from you to stop at.

This is very easy to integrate with any casino engine as it takes a number to stop at. You keep the number generation logic at backend and assign the numbers generated by a casino engine to the dices. Later when all dices are stopped, you compare with the random numbers generated from the game engine to pay for any win to avoid game hacks.

This Dice Class is purely developed on Phaser 3.11 and JavaScript, and no other libraries are being used for any purpose.

**Customizing the dice:**

You must redesign the dice images to give the dice a new look and all other things are being taken care. The image is a sprite of six sides. You can use any format by the way.

**How the Dice Work:**

Dice class allows few parameters

1. size: This is the size of the dice, you can set any number while initializing the Dice Instances.
2. rand: This is a random number at which the dice must be stopped. Pass this number from a server generated instance and verify the same from this game to the server instance for any payout to stop illegal hack.
3. stopped: This is to stop and play the dices.
4. this.quad1 = me.add.quad(40, 40, quads, 0);
5. new Dice(quads1, 145, 100, model.verts); // image to use, position, position y, vertices

It has animate method which calls updateFrameFunc method which generates different sides with transformations applied to each side to make the dice look 3d. The images of each side are being rendered with quad feature of Phaser JS 3.11.

To set delay of rotation, please try with different numbers at this line below,

dice.rotations = { x:dice.finalRot.x-28, y:dice.finalRot.y-28, z:dice.finalRot.z-28};

Browsers Supported:

1. Google Chrome
2. Firefox
3. Opera
4. IE (Game must be served from a server otherwise Dices will not be rendered)

**Features available on demand**

1. Integration to server (PHP or NodeJs)
2. Payout probability calculation
3. Multi player betting games

**Contact Us:**

For any customization or integration related help you can reach out to me at [Mitrabinda.jvmba@gmail.com](mailto:Mitrabinda.jvmba@gmail.com). We are available for freelance work as well for any web related requirements.

Key words:

Phaser 3 3D Dice, Phaser 3 3D Dice Class, Phaser 3D Dice, Customizable Phaser 3d Dice, Phaser 3.11 3D Dice Class, 3d Dice for JavaScript Games