[**https://github.com/nodegui/nodegui-starter**](https://github.com/nodegui/nodegui-starter)

[**https://canvas-gauges.com/documentation/**](https://canvas-gauges.com/documentation/)

[**https://rawgit.com/Mikhus/canvas-gauges/master/examples/radial-component.html**](https://rawgit.com/Mikhus/canvas-gauges/master/examples/radial-component.html)

**https://canvas-gauges.com/documentation/user-guide/advanced-usage**

**https://canvas-gauges.com/documentation/examples/**

[**README.md**](https://github.com/nodegui/nodegui-starter#readme)

**nodegui-starter**

**Clone and run for a quick way to see NodeGui in action.**

**To Use**

To clone and run this repository you'll need [Git](https://git-scm.com) and [Node.js](https://nodejs.org/en/download/) (which comes with [npm](http://npmjs.com)) installed on your computer.

Make sure you have met the requirements listed here: <https://docs.nodegui.org/docs/guides/getting-started#developer-environment>

From your command line:

# Clone this repository

git clone https://github.com/nodegui/nodegui-starter

# Go into the repository

cd nodegui-starter

# Install dependencies

npm install

# Run the app

npm start

**Resources for Learning NodeGui**

* [docs.nodegui.org](https://nodegui.github.io/nodegui) - all of NodeGui and React Desktop's documentation

**Packaging app as a distributable**

In order to distribute your finished app, you can use [@nodegui/packer](https://github.com/nodegui/packer)

**Step 1: (*Run this command only once*)**

npx nodegui-packer --init MyAppName

This will produce the deploy directory containing the template. You can modify this to suite your needs. Like add icons, change the name, description and add other native features or dependencies. Make sure you commit this directory.

**Step 2: (*Run this command every time you want to build a new distributable*)**

Next you can run the pack command:

npm run build

This will produce the js bundle along with assets inside the ./dist directory

npx nodegui-packer --pack ./dist

This will build the distributable using @nodegui/packer based on your template. The output of the command is found under the build directory. You should gitignore the build directory.

More details about packer can be found here: <https://github.com/nodegui/packer>

**License**

MIT

Сделать как здесь с VS2022

<https://learn.microsoft.com/en-us/visualstudio/javascript/tutorial-nodejs-with-react-and-jsx?view=vs-2022>

<https://www.jqueryscript.net/chart-graph/>

<https://webdeveloper.com/tools/>

<https://projects.calebevans.me/jcanvas/docs/retrieveLayers/>

$(document).ready(function(){

            let colors = ["purple", "orange","green", "pink", "cyan", "maroon", "blue", "brown", "grey", "yellow", "black"]

            function DrawCircles() {

                for(let i=0; i<20; i++)

                {

                let diameter = function (min, max) {

                 min = Math.ceil(min);

                 max = Math.flooУфr(max);

                return Math.floor(Math.random()\*(max - min + 1))+ min;

                }

                let randomColor = colors[Math.floor(Math.random()\*colors.length)];

                let color = $('.circles').css("color", randomColor);

                let x = Math.floor(Math.random()\*$window.innerWidth [S]- diameter(50, 200)))+ diameter(50, 200);[/S] // ограничивает площадь экрана для рисования

                let y = Math.floor(Math.random()\*$window.innerHeight [S]- diameter(50, 200))) + diameter(50, 200)[/S];

                let width = diameter(50, 200);

                let circle = $('<div>').addClass('circles').css({'border-radius': '50%', 'background-color': randomColor,

                'position' : 'absolute', 'top': x, 'left': y, 'width': width, 'height': width}); // когда 'width': diameter(50,200) и 'height;:(diameter(50, 200), то значения у них получаются и разные и вместо кругов функция рисует овалы

                $('body').append(circle);

            }

            }

            DrawCircles(20);

        });