Mục Lục

[I. Mô tả mini game Tic Tac Toe. 2](#_Toc507477875)

[II. Cách chơi. 2](#_Toc507477876)

[III. Môi trường cần sử dụng. 2](#_Toc507477877)

[IV. Cách cài đặt game mini caro. 6](#_Toc507477878)

[1. Trang chứa source game Tic Tac Toe. 6](#_Toc507477879)

[2. Cài đặt với notepad++. 7](#_Toc507477880)

[3. Mã code trang. 10](#_Toc507477881)

[4. Chạy trang Tic Tac Toe. 21](#_Toc507477882)

**Hướng dẫn cài đặt Tic Tac Toe**

1. **Mô tả mini game Tic Tac Toe.**

Tic Tac Toa là một mini game được viết bởi mã HTML kết hợp với SCSS, JavaScript. Mini game này giống như cờ caro người chơi tự đánh với mình qua thao tác click chuột vào 9 ô được kẻ sẵn. Các kí hiệu để phân biệt quân cờ “O” và “X”.

1. **Cách chơi.**

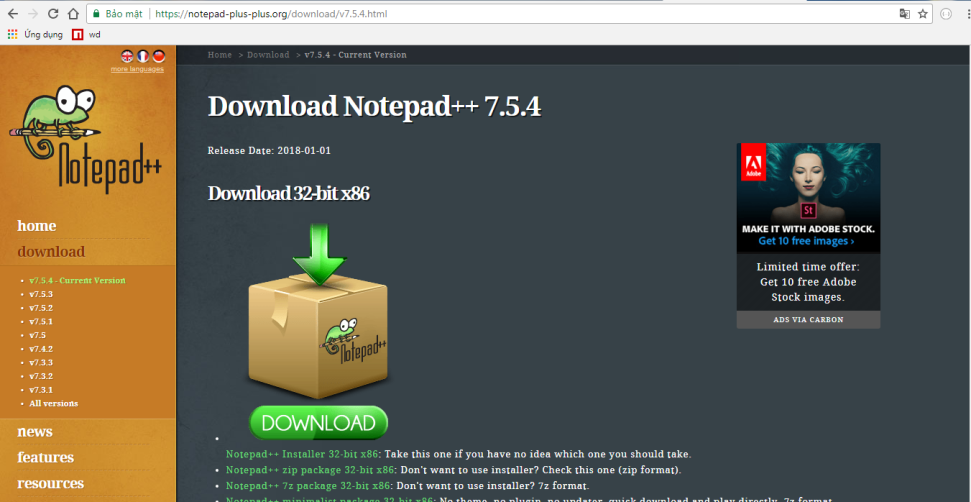
Người chơi nào đạt được một đường có 3 quân giống nhau liên tiếp trước sẽ thắng, ngược lại sẽ thua. Trường hợp 2 bên không có đường thẳng/chéo nào đạt 3 quân liên tục sẽ hòa nhau.

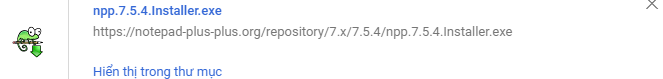
1. **Môi trường cần sử dụng.**

Môi trường cần thiết lập cho game mini này là nodepad++ đây là môi trường cơ bản mà hầu hết mọi người sử dụng. Ngoài ra người dùng có thể sử dụng ngôn ngữ khác để cài đặt game.

1. **Cài đặt nodepad++.**

Download nodepad++ tại trang “<https://notepad-plus-plus.org/download/v7.5.4.html>”.



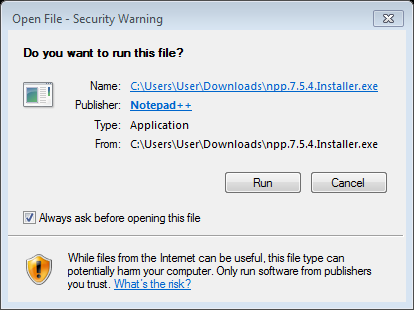


**Các bước cài đặt:**

Sau khi đã tải xong ta tiến hành cài đặt các bước như dưới đây:



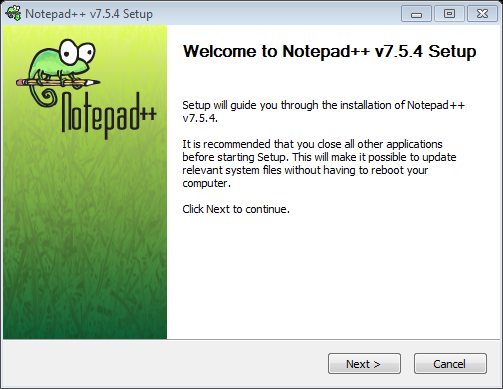
Nhấn “Run”.



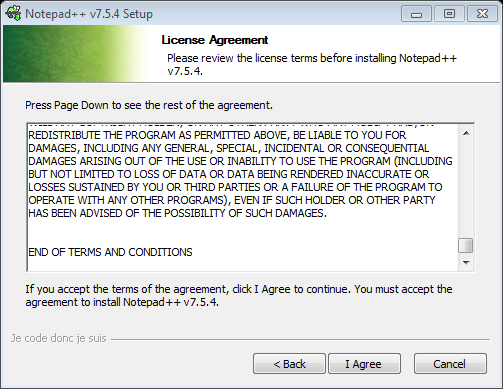
Nhấn “Ok”.



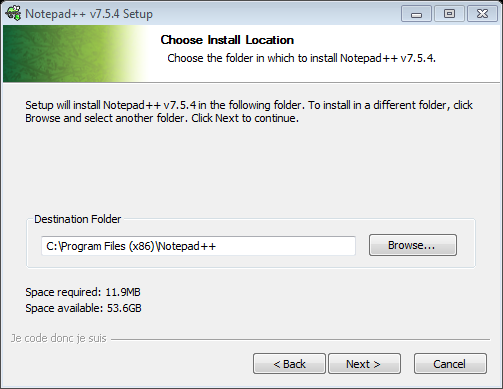
Tiếp tục bấm “Next”.

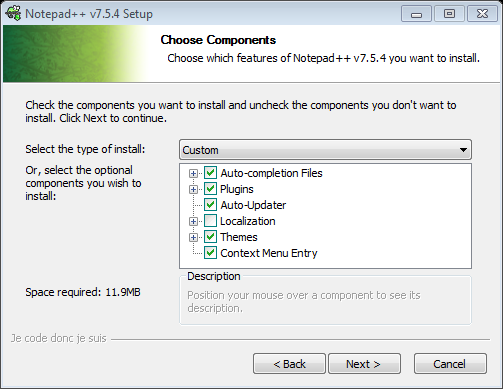


Nhấn “I Agree” để tiến hành cài đặt.

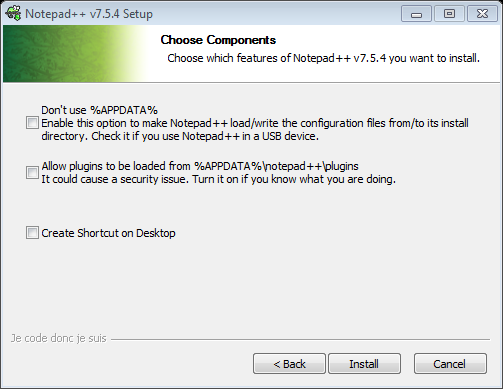


Nhấn “Next”.

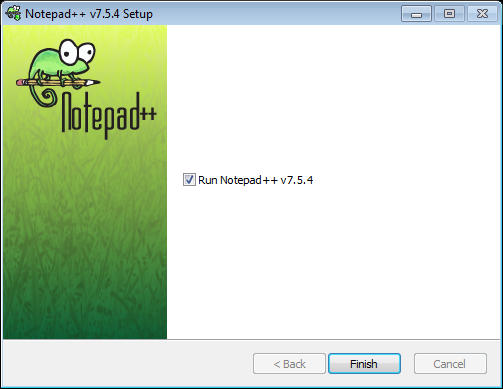




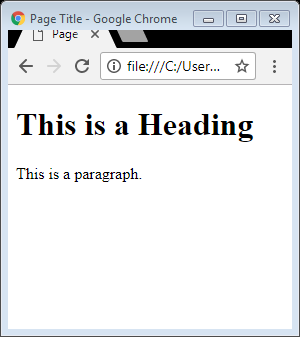
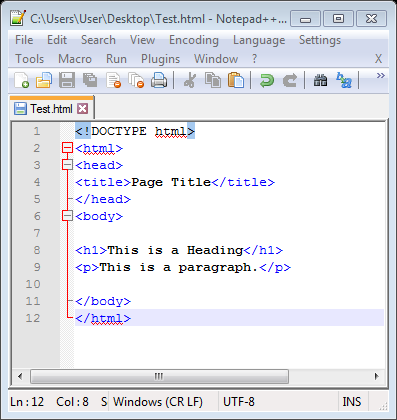
Nhấn “Install” để cài đặt.



Nhấn “” để kết thúc.

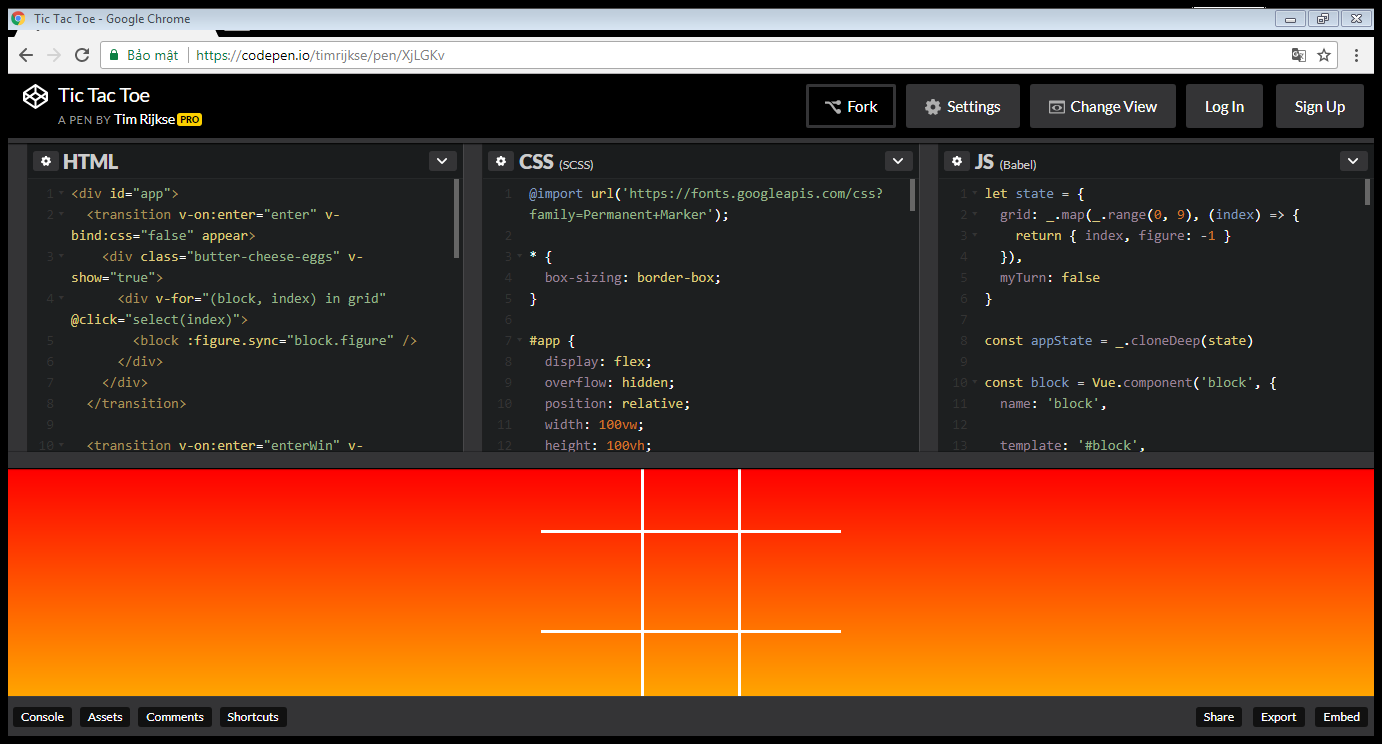


1. **Test thử nodepad++.**

****

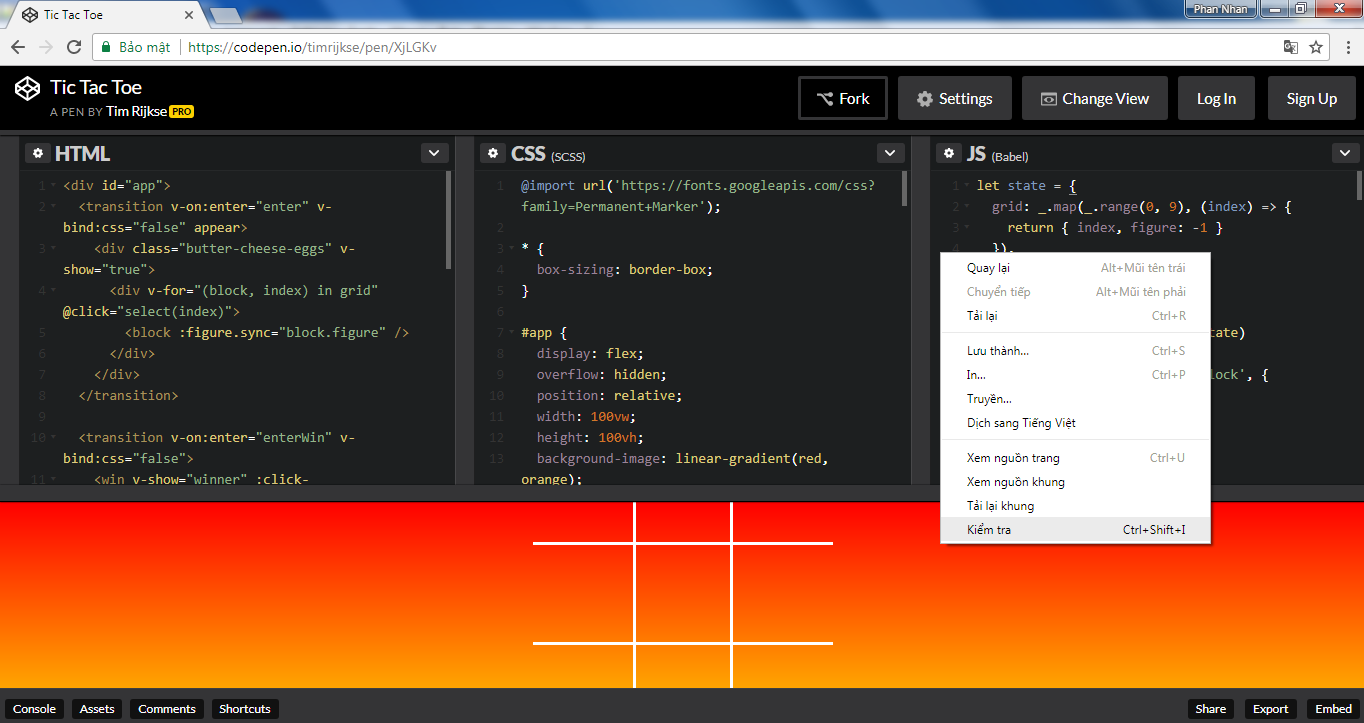
1. **Cách cài đặt game mini caro.**
2. **Trang chứa source game Tic Tac Toe.**

Trang này chứa các mã code: html, css,js, ngoài ra trang có phần demo cho người dùng thực hiện thao tác chơi game.

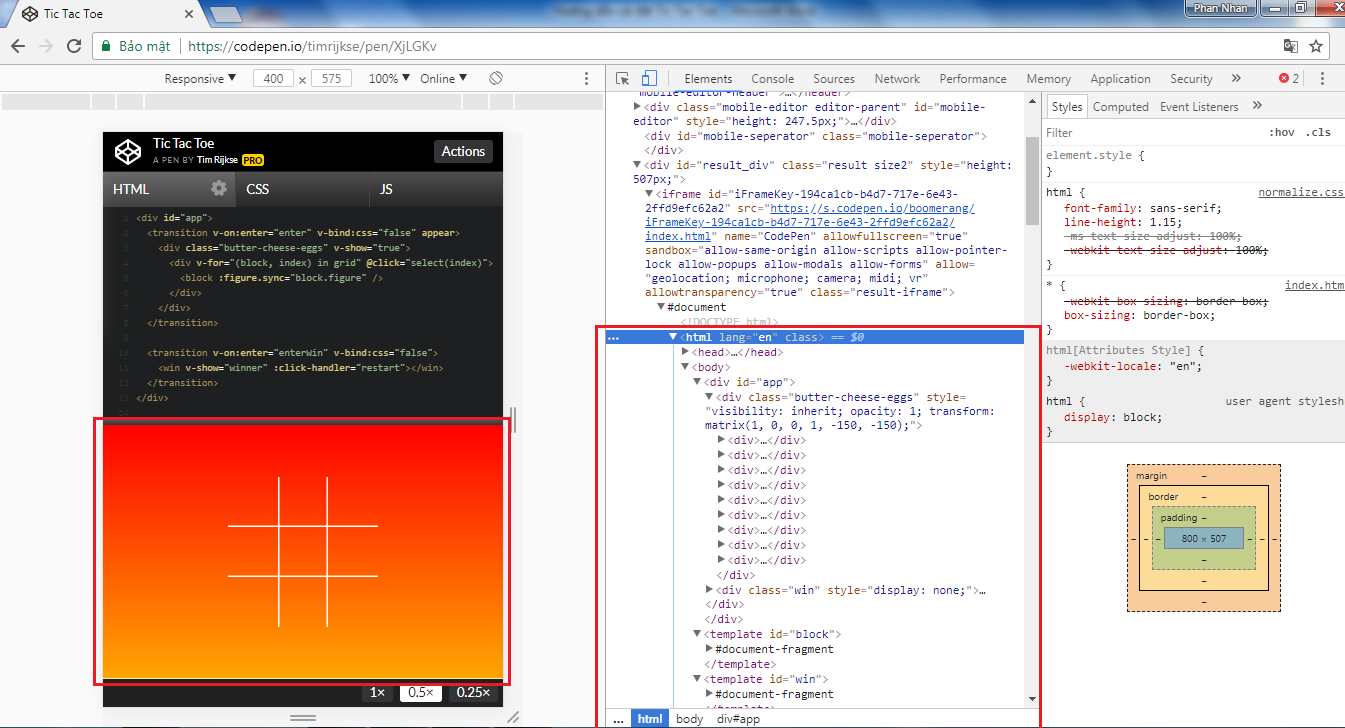


1. **Cài đặt với notepad++.**

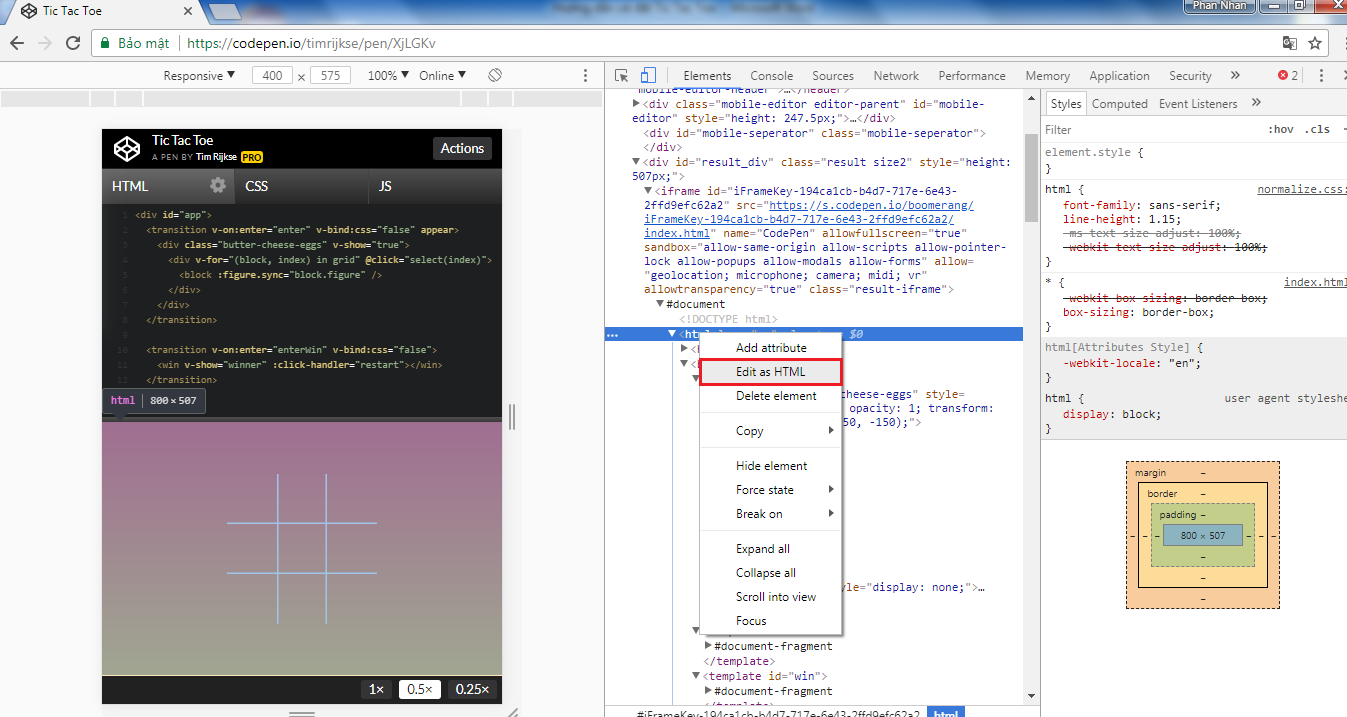
Từ phần code demo thực hiện thao tác chuột với vùng cần lấy mã code.



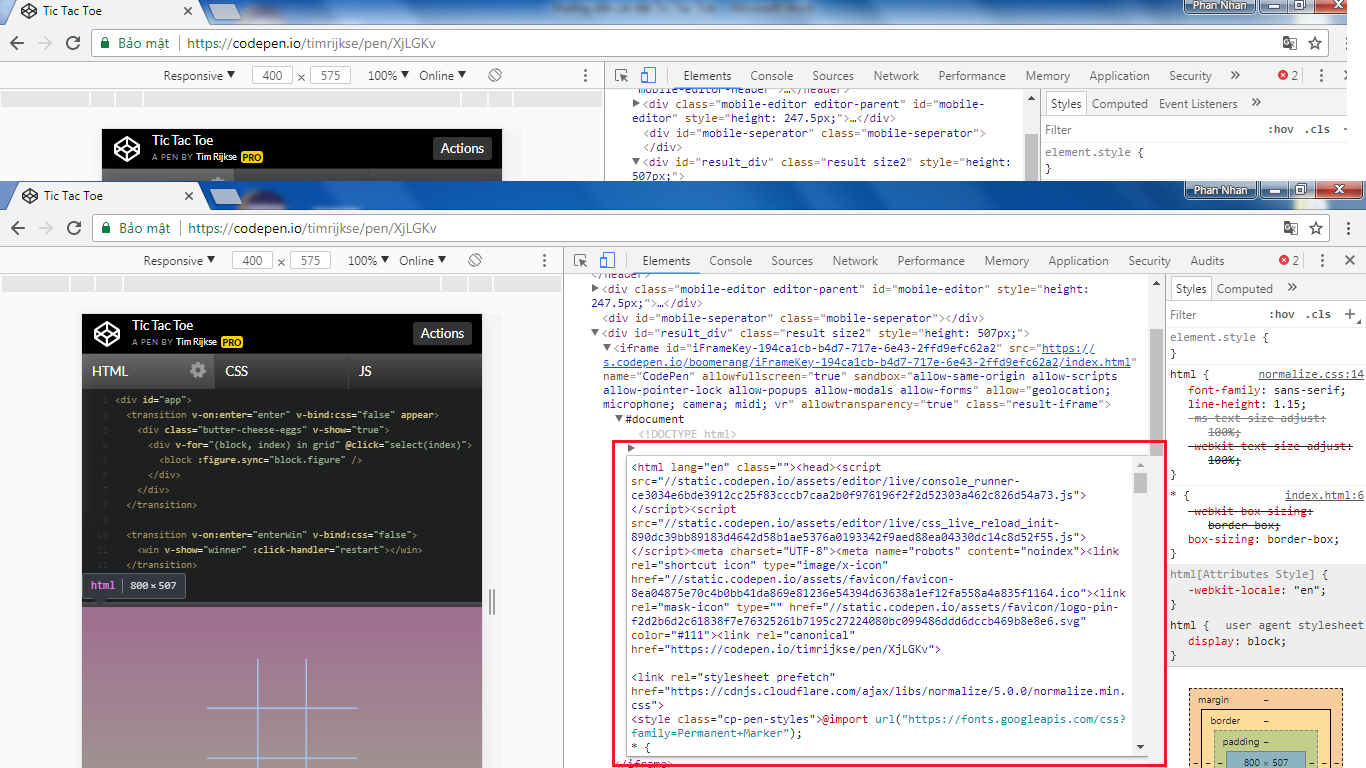
Phần code bên dưới ứng với phần màu cam được khoanh bên trái.



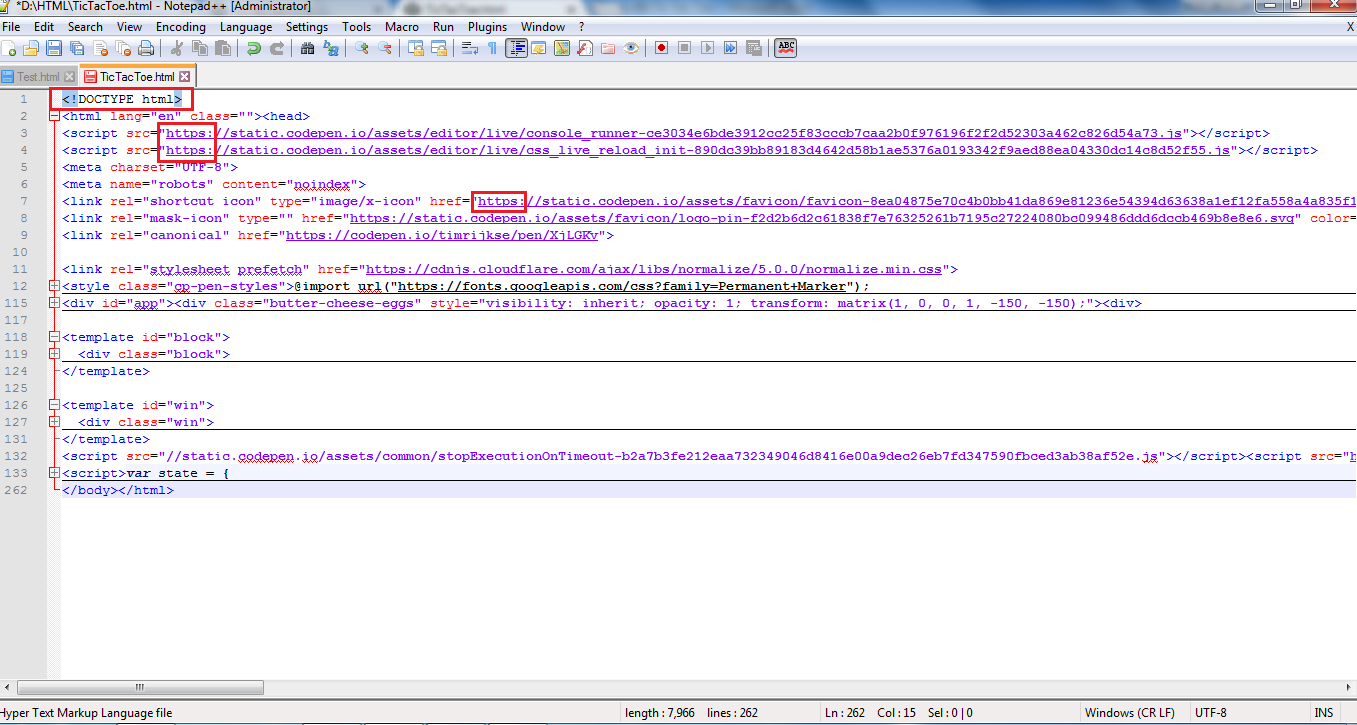
Nhấn chuột vào “ Edit as HTML” để lấy source tương ứng.

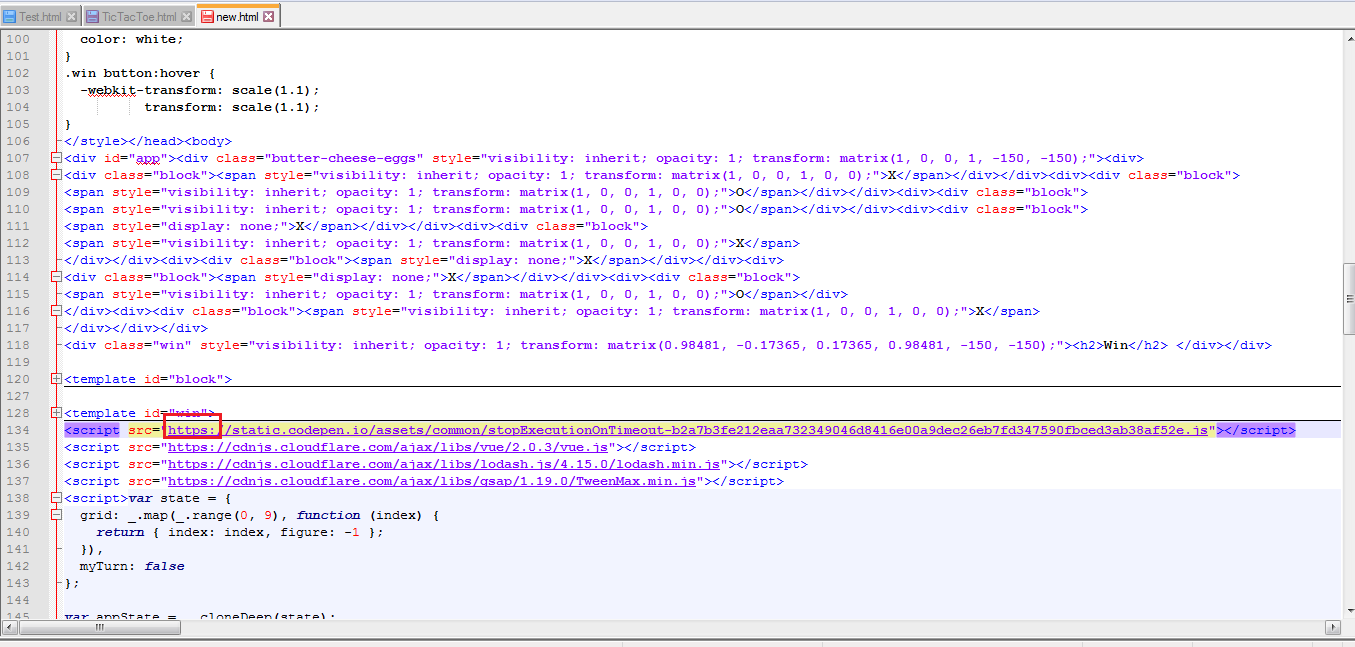


Coppy tất cả phần code ở đây và dán vào notepad++.



Thêm các “https:” vào trước “://static.codepen…” ô khoanh đỏ như bên dưới.

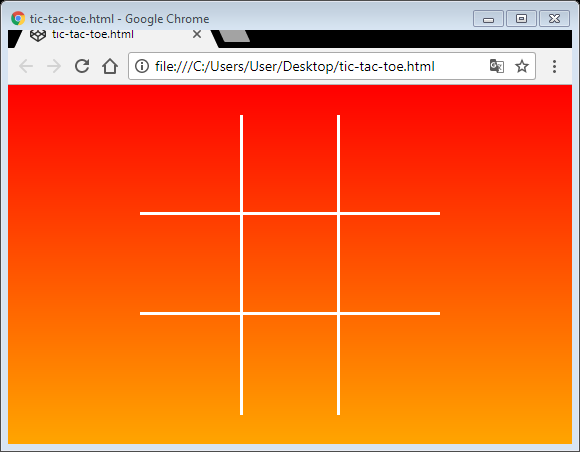


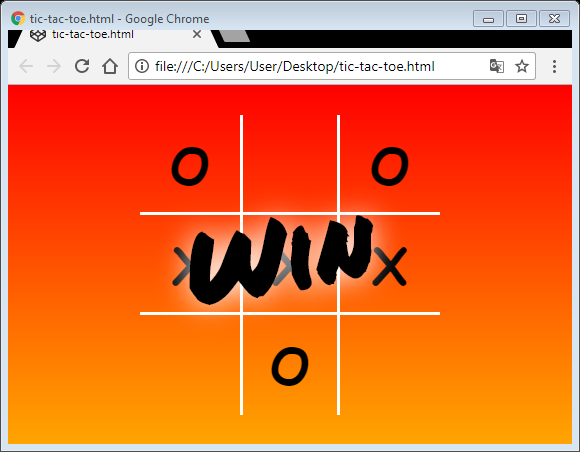


1. **Mã code trang.**

|  |
| --- |
| Tic Tac Toe |
| <!DOCTYPE html>  <html lang="en" class=""><head>  <script src="https://static.codepen.io/assets/editor/live/console\_runner-ce3034e6bde3912cc25f83cccb7caa2b0f976196f2f2d52303a462c826d54a73.js"></script>  <script src="https://static.codepen.io/assets/editor/live/css\_live\_reload\_init-890dc39bb89183d4642d58b1ae5376a0193342f9aed88ea04330dc14c8d52f55.js"></script>  <meta charset="UTF-8">  <meta name="robots" content="noindex">  <link rel="shortcut icon" type="image/x-icon" href="https://static.codepen.io/assets/favicon/favicon-8ea04875e70c4b0bb41da869e81236e54394d63638a1ef12fa558a4a835f1164.ico">  <link rel="mask-icon" type="" href="https://static.codepen.io/assets/favicon/logo-pin-f2d2b6d2c61838f7e76325261b7195c27224080bc099486ddd6dccb469b8e8e6.svg" color="#111">  <link rel="canonical" href="https://codepen.io/anon/pen/yvRmRG">  <link rel="stylesheet prefetch" href="https://cdnjs.cloudflare.com/ajax/libs/normalize/5.0.0/normalize.min.css">  <style class="cp-pen-styles">@import url("https://fonts.googleapis.com/css?family=Permanent+Marker");  \* {  -webkit-box-sizing: border-box;  box-sizing: border-box;  }  #app {  display: -webkit-box;  display: -ms-flexbox;  display: flex;  overflow: hidden;  position: relative;  width: 100vw;  height: 100vh;  background-image: -webkit-gradient(linear, left top, left bottom, from(red), to(orange));  background-image: linear-gradient(red, orange);  }  .butter-cheese-eggs {  position: absolute;  left: 50%;  top: 50%;  -webkit-transform: translate(-50%, -50%);  transform: translate(-50%, -50%);  display: -webkit-box;  display: -ms-flexbox;  display: flex;  -ms-flex-wrap: wrap;  flex-wrap: wrap;  width: 300px;  height: 300px;  }  .butter-cheese-eggs > div {  -webkit-box-flex: 0;  -ms-flex: 0 0 auto;  flex: 0 0 auto;  width: 100px;  height: 100px;  }  .butter-cheese-eggs > div .block {  cursor: pointer;  display: table-cell;  width: 100px;  height: 100px;  font: bold 50px/0 'Comic Sans MS', sans-serif;  vertical-align: middle;  text-align: center;  }  .butter-cheese-eggs > div:nth-child(1), .butter-cheese-eggs > div:nth-child(2), .butter-cheese-eggs > div:nth-child(3), .butter-cheese-eggs > div:nth-child(4), .butter-cheese-eggs > div:nth-child(5), .butter-cheese-eggs > div:nth-child(6) {  border-bottom: 3px solid #fff;  }  .butter-cheese-eggs > div:nth-child(2), .butter-cheese-eggs > div:nth-child(5), .butter-cheese-eggs > div:nth-child(8) {  border-left: 3px solid #fff;  border-right: 3px solid #fff;  }  .butter-cheese-eggs span {  display: block;  }  .win {  position: absolute;  left: 50%;  top: 50%;  -webkit-transform: translate(-50%, -50%) rotate(-10deg);  transform: translate(-50%, -50%) rotate(-10deg);  -webkit-transform-origin: 50% 50%;  transform-origin: 50% 50%;  -webkit-transform-origin: center;  transform-origin: center;  width: 300px;  height: 300px;  margin-top: -60px;  margin-left: -20px;  padding-top: 140px;  font-family: 'Permanent Marker', cursive;  text-align: center;  }  .win h2 {  display: block;  margin: 0 0 10px;  font-size: 100px;  text-shadow: 0px 0px 50px white;  text-align: center;  }  .win button {  cursor: pointer;  -webkit-transition: all 200ms cubic-bezier(0.175, 0.885, 0.32, 1.275);  transition: all 200ms cubic-bezier(0.175, 0.885, 0.32, 1.275);  outline: none;  display: inline-block;  padding: 15px;  background-color: darkred;  border: none;  border-radius: 10px;  font-family: 'Permanent Marker', cursive;  font-size: 20px;  color: white;  }  .win button:hover {  -webkit-transform: scale(1.1);  transform: scale(1.1);  }  </style></head><body>  <div id="app">  <transition v-on:enter="enter" v-bind:css="false" appear>  <div class="butter-cheese-eggs" v-show="true">  <div v-for="(block, index) in grid" @click="select(index)">  <block :figure.sync="block.figure" />  </div>  </div>  </transition>    <transition v-on:enter="enterWin" v-bind:css="false">  <win v-show="winner" :click-handler="restart"></win>  </transition>  </div>  <template id="block">  <div class="block">  <transition v-on:enter="enter" v-bind:css="false">  <span v-show="figure > -1">{{ fig }}</span>  </transition>  </div>  </template>  <template id="win">  <div class="win">  <h2>Win</h2>  <!--<button @click="clickHandler">Play again</button>-->  </div>  </template>  <template id="block">  <div class="block">  <transition v-on:enter="enter" v-bind:css="false">  <span v-show="figure > -1">{{ fig }}</span>  </transition>  </div>  </template>  <template id="win">  <div class="win">  <h2>Win</h2>  <!--<button @click="clickHandler">Play again</button>-->  </div>  </template>  <script src="https://static.codepen.io/assets/common/stopExecutionOnTimeout-b2a7b3fe212eaa732349046d8416e00a9dec26eb7fd347590fbced3ab38af52e.js"></script>  <script src="https://cdnjs.cloudflare.com/ajax/libs/vue/2.0.3/vue.js"></script>  <script src="https://cdnjs.cloudflare.com/ajax/libs/lodash.js/4.15.0/lodash.min.js"></script>  <script src="https://cdnjs.cloudflare.com/ajax/libs/gsap/1.19.0/TweenMax.min.js"></script>  <script>var state = {  grid: \_.map(\_.range(0, 9), function (index) {  return { index: index, figure: -1 };  }),  myTurn: false  };  var appState = \_.cloneDeep(state);  var block = Vue.component('block', {  name: 'block',  template: '#block',  props: {  figure: {  type: Number,  default: -1  }  },  computed: {  fig: function fig() {  return this.figure === 0 ? 'O' : 'X';  }  },  data: function data() {  return {  selected: false  };  },  methods: {  enter: function enter(el, done) {  TweenMax.from(el, 1, {  autoAlpha: 0,  scale: 0,  ease: Elastic.easeOut.config(1.25, 0.5),  onComplete: done  });  }  }  });  var win = Vue.component('win', {  name: 'win',  template: '#win',  props: {  clickHandler: {  type: Function,  default: null  }  }  });  var app = new Vue({  name: 'app',  el: '#app',  data: function data() {  return state;  },  components: {  block: block  },  computed: {  winner: function winner() {  var wins = ['012', '036', '345', '147', '258', '678', '048', '246'];  var grid = this.grid;  var player = this.myTurn ? 0 : 1;  var moves = \_.reduce(this.grid, function (result, value, index) {  if (value.figure === player) {  result.push(index);  }  return result;  }, []);  return !!\_.find(wins, function (win) {  var combination = \_.map(win.split(''), function (n) {  return parseInt(n);  });  console.log('combination', combination, moves);  return \_.difference(combination, moves).length === 0;  });  }  },  methods: {  select: function select(index) {  var figure = this.grid[index].figure;  if (figure > -1) {  return;  }  this.grid[index].figure = this.myTurn ? 1 : 0;  this.myTurn = !this.myTurn;  },  restart: function restart() {  this.grid = appState.grid;  this.myTurn = appState.myTurn;  },  enter: function enter(el, done) {  TweenMax.from(el, 1, {  autoAlpha: 0,  scale: 0,  ease: Elastic.easeOut.config(1.25, 0.5)  });  },  enterWin: function enterWin(el) {  TweenMax.from(el, 1, {  autoAlpha: 0,  scale: 0,  ease: Elastic.easeOut.config(1.25, 0.5)  });  }  }  });  //# sourceURL=pen.js  </script>  </body></html> |

1. **Chạy trang Tic Tac Toe.**

****

****