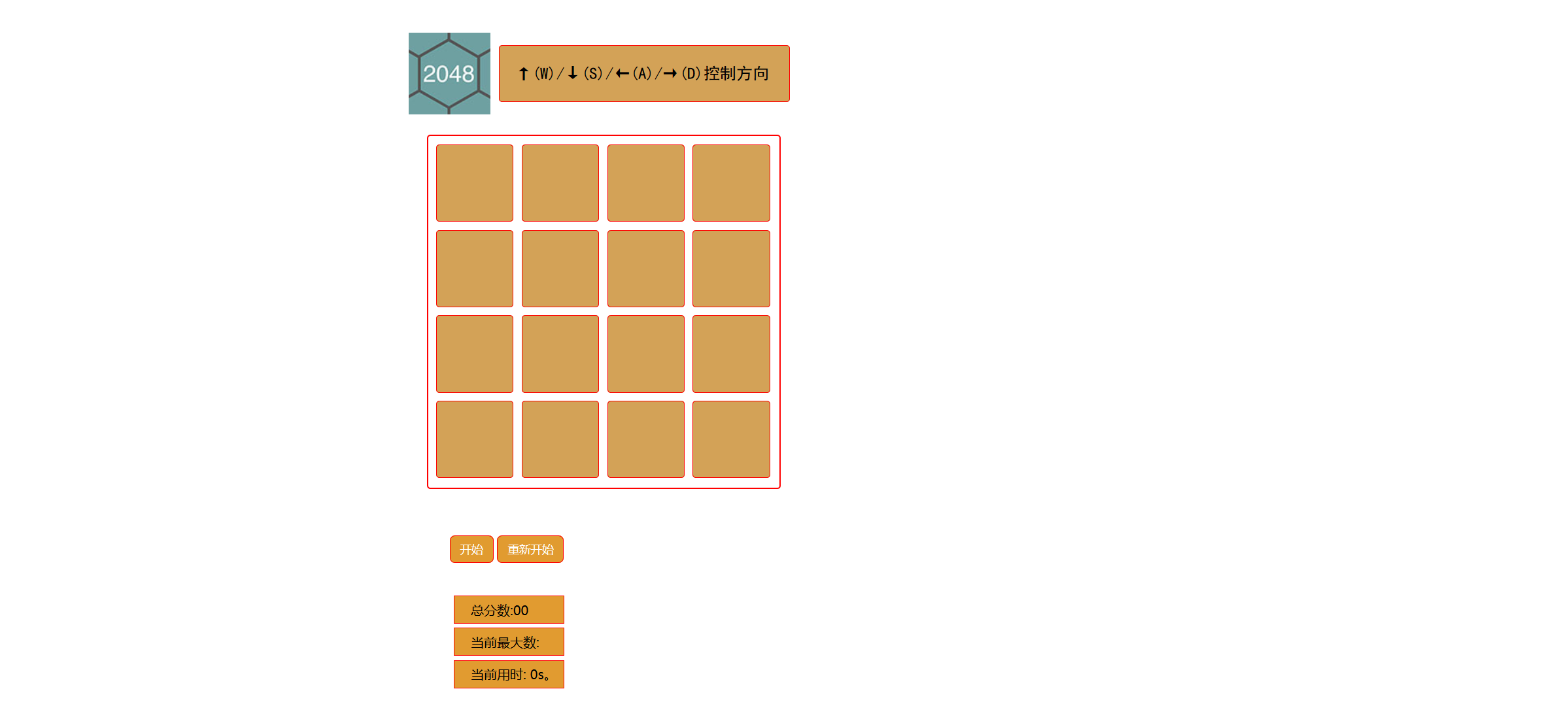
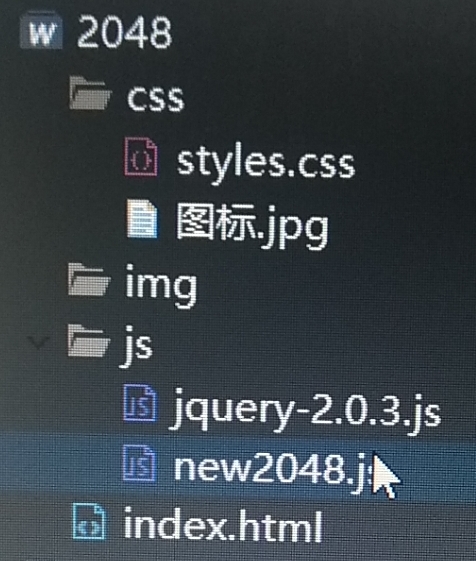
五.详细设计

1.游戏主页面

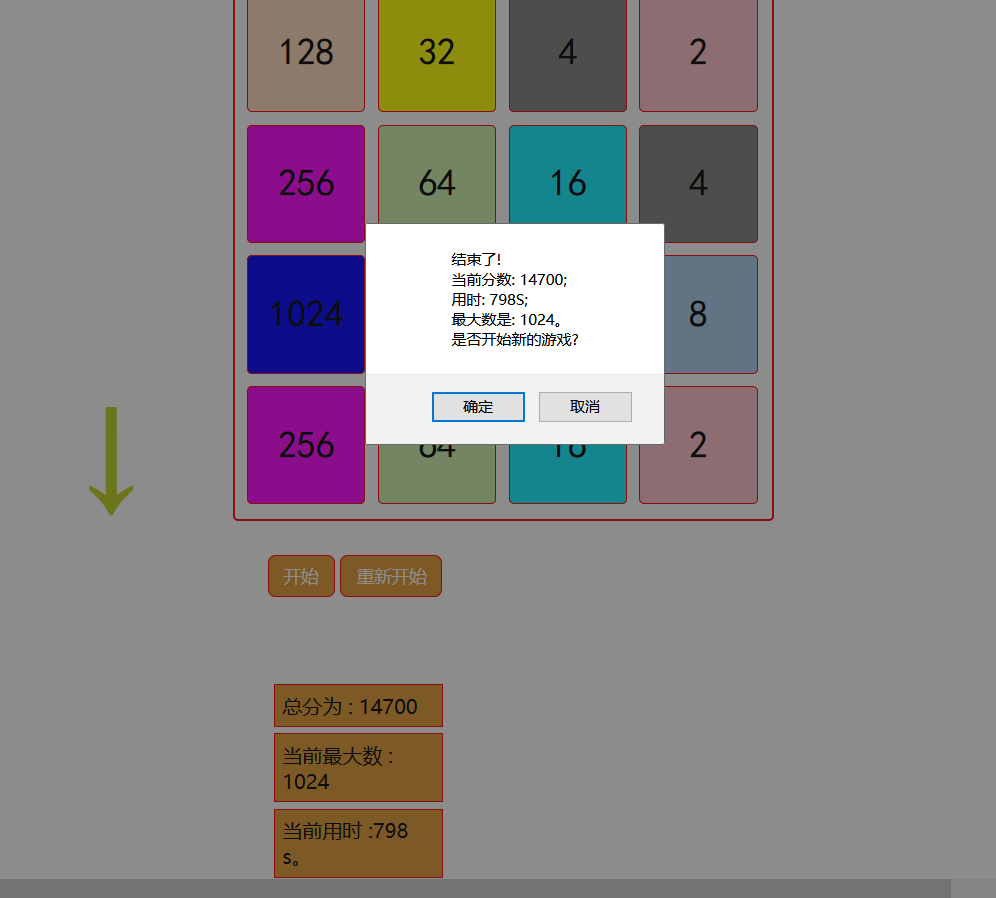
截图如下：



2，该项目的主要结构如下：



3.游戏结束：



4.简单的流程图

是

移动获得分数

两个数相加，暂时加和

并保存最大的数

是

游戏结束

否

注意了哦

否

是

是否有空格？

同一方向上是否相加

↑(W)/↓(S)/←(A)/→(D)控制方向移动

4\*4方格随机产生两个数字

游戏开始

4相关代码设计：

（1）初始化好定义的变量 keys colors score max time t

（2）用window.onkeydown = function(e) 来获得键盘按下的事件

（3）用function isEnd()判定结束时的成绩

function isEnd() {

var f = false;

if (locations.indexOf(0) == -1) {

$("#danger").text("注意了哦~");

{if (isEndX() && isEndY()) {

clearTimeout(t);

if (window.confirm("结束了!\n当前分数: " + score + ";\n用时: " + time

+ "S;\n最大数是: " + max + "。\n是否开始新的游戏?")) {

init();

} else {window.close();}

}

} else {$("#danger").text("");

}return f;}

（4）function isEndX() / function isEndY()判断横向/纵向的数组，如果相邻位置的数不相同,就结束

unction isEndX() {

var f = false;

var w = new Array();

for (var j = 0; j < 4; j++) {

for (var i = 0; i < 4; i++) {

w[i] = locations[4 \* j + i];}

f = (w[0] != w[1] && w[1] != w[2] && w[2] != w[3]);

if (!f) {

break;

}

}

return f;

}

（5）创建向四个方向移动相加的方法

Eg：（向下）

function toDown() {

for (var i = 0; i < 4; i++) {

var row = [ locations[i + 12], locations[i + 8], locations[i + 4],

locations[i] ];

configurationD(i, row); }

locations[createLocation()] = createFixedNum();

paint();}

（6）用function paint()来计算分数和最大数

function paint() {

for (var i = 0; i < 16; i++) {

$("#box" + keys[i]).text((locations[i] == 0) ? "" : locations[i]);

var index = (locations[i] == 0) ? 0

: (locations[i].toString(2).length - 1);

$("#box" + keys[i]).css("background", colors[index]);

if (locations[i] > max) {

max = locations[i];}

}

$("#score").text("总分为 : " + score);

$("#max").text("当前最大数 : " + max);

}

（7）在空位置中随机生成2和4 ，生成2的概率为0.8。

在空位置中随机生成，如果该位置有值,就返回重新生成

function createFixedNum() {

return Math.random() < 0.8 ? 2 : 4;

}

function createLocation() {

var num = Math.floor(Math.random() \* 16);

while (locations[num] != 0) {

num = Math.floor(Math.random() \* 16);

}

return num;

}

（8）用function showtime()来记时

function showtime() {

$("#time").text("当前用时 :" + (++time) + " s。");

}