**软件功能说明书 (Functional Spec)**

1. 对相关的概念的定义

随机布雷 ：可以由电脑进行随机布雷

自定义雷的数量 ：可以由用户自定义雷的数量

计时 ：有计时装置，可以自开始扫雷后自动记时

游戏结束提示： 在游戏结束时会出现提示对话框，并停止计时

自定义雷的数量界面： 自定义雷的数量界面

程序主界面： 程序运行的主界面

定义鼠标左键为点击，点击未知出先数字、空白或地雷。鼠标右键为标志，标志处形成旗帜。

1. 规范假设前提

程序需有一定的硬件运行环境，能在相应的软件上成功的运行。

1. 描述主流的用户/软件交互步骤

(1) 扫雷游戏分为初级、中级和高级三个级别。单击游戏菜单可以选择“初级”、“中级”和“高级”。

(2) 选择级别后将出现相应级别的扫雷区域，这时用户使用鼠标左键单击‘“开始”便启动计时器。

(3) 用户要揭开某个方块，可左键单击它。若所揭方块下有雷，用户便输了这一局，若所揭方块下无雷，则显示一个数字，该数字代表方块的周围的8个方块中共有多少颗雷。

4) 如果用户认为某个方块下埋着雷，单击右键可以在方块上标识一个用户认为是雷的图标，即给出一个扫雷标记。用户每标记出一个扫雷标记(无论用户的标记是否正确)，程序将显示的剩余雷数减少一个。

(5)当用户点中雷时，将有“你输了”的提示出现，胜利时，将有“你赢了”的提示出现。

1. 功能的某些副作用

点击打开未知。显示周围雷的数量。如果雷的数量为0。则用一个递归计算这个雷周围8个格子里有没有周边雷数量为0的格子，如果有，再计算这个格子周围。直到把相连的所有周边雷数量为0的格子找出来，放入一个数组。然后再把这个数组里所有格子以及它们周围的8个格子显示出来，显示每个格子周围雷的数量 。雷的数量用不同颜色。算法效率不是很高。被打开的格子或被标志的格子，没有hover事件。

1. 服务质量的说明。

该程序可以实现难度等级分级，分别为初级、中级和高级，并且可以实现用户自定义雷的数量。可以满足不同用户的需求，并且为用户提供由简入难的游戏体验，并且游戏简单，上手较快，对思维方式与计算方式有一定的训练帮助，能较好的满足用户对扫雷游戏的需求。

**软件技术说明书（设计文档）**

1. 软件如何实现抽象 ( Abstraction )

 var mine1

window.onload = function(){

document.getElementsByClassName('mine-wrap')[0].style.display = "block";

mine1 = new Mine("mine1",'face',30,16,99,"game-tag-images","game-time-images");

mine1.init();

var inputEle = document.getElementsByTagName('input');

对地雷以及棋盘进行抽象定义。在抽象的最高层次使用问题环境的语言，在低层次上使用更过程化的方法，把面向问题的术语和面向实现的术语结合起来描述

1. 内聚／耦合 ／模块化 是如何实现的

内聚是衡量模块内成分关联的“单位”。 偶然内聚：一个模块内的各部分没有任何联系，就叫偶然内聚。例如一个程序，我就用一个类来写可不可以？可以，什么变量，函数都塞这个类里，今天写完明天来找保证你自己都头大，更别说项目交付给别人来管理了。逻辑内聚：一个模块内的各部分存在逻辑上的联系，通过参数不同来实现不同功能。这种做法也比较蠢，时间内聚：需要在某个时间段内执行的方法放到一起，例如在场景加载时需要处理的一些数据。过程内聚：在一个过程中，你需要按一定的顺序来执行你的代码块。ABCDEFG这样下去的，把他们写到一起，看起来很有顺序感。通信内聚：指模块内所有处理元素都在同一个数据结构上操作或所有处理功能都通过公用数据而发生关联（有时称之为信息内聚）。即指模块内各个组成部分都使用相同的数据数据或产生相同的数据结构（来自百度）。自己翻译一遍就是，一团面（数据），你要用来包饺子下面条还是当橡皮泥玩，只要是处理这团面的，那就全写一起。顺序内聚：和过程内聚有点相似，和一个功能密切相关，A输出的参数为B传入的参数，B输出的参数为C传入的参数，那把他们都写一起。功能内聚：一个模块内的所有元素，都是为了处理一个功能的，这是最强的内聚。例如，在3D场景中，我们需要播放声音，那就把所有有关声音的东西写在一起，你需要播放什么声音的时候告诉我一声，我来播放就好。

3、信息是如何隐藏和封装的

function foo() {

var a = 10;

function bar() {

a \*= 2;

return a;

}

return bar;

}

var baz = foo();

baz();*//return 20*

baz();*//return 40*

baz();*//return 80*

var blat = foo();

blat();*//return 2*

这里我们并不能通过foo.a访问a，因为a并不是foo的属性，只是在该函数作用域内声明的一个局部变量。而当我们调用bar函数时，该函数并非运行在其调用时所在作用域，而是运行在定义bar函数的作用域处，即a所在作用域，故bar函数作为foo的子函数，可以访问其父函数中的变量，并且不同的实例化对象，各自有一个作用域和变量和子函数的副本，不同实例中的变量值无关。总结而言

将想隐藏的信息使用var声明而非使用this.变量名声明。

通过定义子函数调用想隐藏的信息（这种用来访问私有变量的函数，又称为特权数）。

不同实例化对象有着不同的作用域、子函数、变量的副本。

4、如何处理错误情况

|  |
| --- |
| Mine.prototype = { |
|  | init:function(){//画格子 |
|  | this.ele.width = this.PANE\_SIZE \* this.panewidth; |
|  | this.ele.height = this.PANE\_SIZE \* this.paneheight; |
|  | this.faceele.src = "res/face\_normal.bmp" |
|  |  |
|  | this.oldPos = [0,0];//鼠标上一个停留的位置。默认值。用于处理hover事件。 |
|  | this.cellArr=[];//格子信息保存数组。保存每个格子是不是雷，当前是否有标记 |
|  | this.mineArr=[];//地雷位置数组。 |
|  | this.time = 0; |
|  | this.notTaged = this.minenum; |
|  | this.numToImage(this.notTaged,this.tagele); |
|  | this.numToImage(this.time,this.timeele); |
|  | this.mousedownArr=''; |
|  | this.createCells();//初始化cellArr数组,并涂上颜色。 |
|  | this.inited = false; |
|  | clearInterval(this.timer); |
|  | this.onmousemove();//鼠标在上面移动，触发每个格子的 |
|  | this.onmouseout();//鼠标移出canvas的事件。 |
|  | this.onmousedown(); |
|  | this.onclick();//点击方格事件 |
|  | this.preRightMenu();//阻止右键菜单。 |
|  | }, |

解决了canvas在高分屏缩放150%之后坐标计算不精确的问题。

5  界面和实现的分离是如何实现的

<html>

<body>

<div>

<div tag="title">Loading...</div>

<table>

<tbody tag="items">

<tr tag="+item\_a"><td>type A</td></tr>

<tr tag="+item\_b"><td tag="title">type B</td></tr>

</tbody>

</table>

</div>

<div tag="arr"></div>

<div tag="arr"></div>

<div tag="arr"></div>

</body>

</html>

那么这个时候打开这个html，我们就可以直接看到它的效果，使用petal-ui-template以后我们可以直接

$(control.dom.title).text("这是一个标题");

$(control.dom.arr[1]).text("这是标记为arr的第二个div。");

至此html设计就可以和js代码分离了，所有的if，for还是放在js端，不要出现在html里。

6 软件是如何保证应对变化的灵活性

提高代码的灵活度，测试和开发同时进行。编写一些代码，马上进行测试和构建。接着，编写更多的代码，继续测试。更好的是，在你编码的时候或者编码之前，就计划好你的测试。测试不是一个独立分开的过程，它是开发的一部分。质量不等同于测试；要想有高质量的产品，就要把开发和测试紧密捆绑在一起，直到不分彼此。

7 如何保持对大量数据的处理能力

HTML静态化   
效率最高、消耗最小的就是纯静态化的html页面，所以尽可能使网站上的页面采用静态页面来实现，这个最简单的方法其实也是最有效的方法。但是对于大量内容并且频繁更新的网站，无法全部手动去挨个实现，于是出现了常见的信息发布系统CMS，像常访问的各个门户站点的[新闻频道](https://www.baidu.com/s?wd=%E6%96%B0%E9%97%BB%E9%A2%91%E9%81%93&tn=SE_PcZhidaonwhc_ngpagmjz&rsv_dl=gh_pc_zhidao)，甚至他们的其他频道，都是通过信息发布系统来管理和实现的，信息发布系统可以实现最简单的信息录入自动生成静态页面，还能具备频道管理、权限管理、自动抓取等功能，