

网络版炸弹人游戏的实现

|  |  |
| --- | --- |
| **专业** | **网络工程** |
| **学号** | **173401040213** |
| **姓名** | **姚茗瀚** |
| **指导教师** | **石祥滨** |
| **负责教师** |  |

2020年6月

摘要

计算机游戏行业蓬勃发展，市场上出现了巨大的缺口，与之对应的许多高校学生开始开发游戏，游戏分为很多种，平台也多种多样，本人开发了一款游戏，游戏平台为web平台，使用了cocoscreator游戏引擎，因为本款游戏是多人联机的，所以需要解决如何能够保证在低延迟，低带宽的情况下能保证有较高的流畅度。

游戏角色是一名医护人员，玩家将操控角色对抗病毒，游戏开发期间，正值新冠疫情的时代主题，因此我们将主题设置为医护人员和病毒之间的抗争，正符合课设期间的实事。

游戏具有匹配，对战，排名，注册，登陆的功能，游戏可以通过手机号短信验证码的方式进行注册和登录，因为游戏采用P2P的对战方式，游戏在匹配时并没有中心服务器，所以需要一种算法来竞选出房主，我采用的是房主竞选算法，游戏在开始匹配阶段时，会先竞选出房主，并由房主来生成地图，生成的地图再发送给另一方，以此来保证在生成地图阶段地图的唯一性，并且当游戏在遭遇较大网络波动的时候会采用自适应算法对网络进行优化，来保证游戏数据的稳定性，自适应算法具体为帧同步加方向预测算法，同时游戏接入了多种系统，包括支付系统，和短信系统，使玩家的游玩儿体验得到保障。

游戏采用了低耦合设计，游戏分为运行时和登录时，二者的耦合度极低，运行时游戏采用了MVVM架构，此架构具有极地的耦合率，所以游戏的匹配速度极快，无需传送无用数据，以此保障了游戏的本地运行性能，同时联机时采用自适应算法，使游戏性能得到极大的提高。

关键词：双人对战；随机地图；角色扮演；匹配算法；自适应算法；

Title of Paper (in English)

**Abstract**

With the rapid development of the computer game industry, there is a huge gap in the market. Many college students begin to develop games. There are many kinds of games and various platforms. I developed a game. The game platform is web platform and uses cocos creator game engine, because this game is online, So we need to solve how to ensure the high fluency in the case of low delay and low bandwidth.

The game player is a medical care worker, and the game player will control the role of the virus. During the game development, it is the theme of COVID-19's era. Therefore, we set the theme as a struggle between medical staff and viruses, which is in line with the actual events during the course of the lesson. It aims to publicize the determination of the whole country to fight COVID-19 and make the game more attractive to the players. At the same time, we will donate the income from recharge the cure coin to the medical staff who are fighting against the epidemic.

The game has the functions of matching, fighting, ranking, registering and logging in. The game can be registered and logged in through SMS verification code of mobile phone number. Meanwhile, the player's data will be saved in the database, so the player can also retrieve the password through mobile phone number. At the same time, the game provides combat service, players can choose the difficulty of combat, and players with the same difficulty can be matched together. At the same time, when the game starts, the game system will automatically optimize according to the player's network to ensure the fluency of the character's action, and the game has the ability of automatic error correction, It will automatically correct to ensure the correct transmission of data, which not only realizes the efficient transmission of data, but also realizes the accurate transmission of data.

**Keywords:Double player; Random map; cosplay; Matching algorithm; War service;**

目录

1 前言 1

1.1 Web游戏背景 1

1.2 Web游戏开发方式 1

1.3 游戏引擎 1

1.4 技术难点 2

2 需求分析 3

2.1 游戏整体需求 3

2.2 系统功能需求介绍 3

2.2.1人物模块需求 3

2.2.2地图模块需求 3

2.2.3炸弹模块需求 3

2.2.4道具模块需求 4

2.2.5匹配模块需求 4

2.2.6排行榜模块需求 4

2.2.7治愈币模块需求 4

2.2.8支付模块需求 4

2.3 游戏架构需求分析 4

2.4 游戏变量需求分析 5

2.5 游戏数据库需求分析 6

2.6 相关技术介绍 7

2.6.1 cocos游戏引擎 7

2.6.2 mysql数据库 7

2.6.3 Servlet 和 JSP 技术 8

3 系统设计 9

3.1 游戏整体架构分析 9

3.2 网络通信架构分析 11

4 详细设计 13

4.1 登陆时系统设计 13

4.2 运行时系统设计 15

4.2.1人物系统 16

4.2.2地图系统 17

4.2.3道具系统 17

4.2.4匹配系统 18

4.3数据结构系统设计 18

4.4坐标系统设计 18

4.5房主竞选系统设计 19

5 测试及运行 21

5.1 游戏开始界面截图展示 21

5.2 技能介绍截图界面展示 22

5.3 双人联机过程截图界面展示 23

5.4 其中一方掉线截图界面展示 25

5.5 分出胜负截图界面展示 26

6 总结 28

参考文献 29

致谢 30

2. 前言
   1. Web游戏背景

在当今世界，现代的web已经慢慢发展成一个可靠的平台，它的功能之多，以至于人们不仅仅想用其进行普通的开发，它不仅能够承载众多高质量的游戏，同时也可以用来宣传和推广这些高质量的游戏。而且采用现代网页和尖端的技术，完全有可能做出一些让人印象深刻的高质量web游戏，其性能和玩儿法以及游戏种类甚至可以和许多桌面级游戏媲美。其主要得益于JavaScript的高性能技术，最近几年，JavaScript的实时编译技术性能得到了大幅度的升级，加上其又开放了很多最新的API，web的游戏性能又得到了进一步的提升。

在当今互联网环境，中国网页游戏的海外市场规模的增长率要远高于国内市场，所以海外市场还有很大的潜力。在当今互联网环境，网页游戏呈现出了“高投入高产出”的状态，这一趋势冲击了互联网原油的生态环境，用户流量仍然是网页游戏的获取用户的重要手段。但是方式更有多样性，比如弹窗广告，贴片广告依然是主要导量形式，但是明星代言和IP改编的用户导入也越来越多的被采用。所以Web游戏发展呈现逐年增长的趋势，而且随着人们生活水平节奏的加快，人们越来越需要一种轻量级，快速的游戏方式，无需繁琐的游戏规则，因此Web游戏脱颖而出，但是Web游戏虽然加载容易但是有着不稳定的缺点，对网络环境依赖性较大，所以如何优化网络环境和采用合适的对战算法便成为了当务之急。

* 1. Web游戏开发方式

Web游戏有多种开发方式，而说到web游戏开发，便不得不说到web前端。Web前端有很大一部分的工作集使用了众多主流的前端框架，例如Vue，React， Angular，虽然种类众多，可供选择的实现方式多种多样，但是不同的抽象方式和实现差距巨大。所以web游戏开发需要一套统一的开发工具，游戏引擎就应运而生了。

* 1. 游戏引擎

游戏引擎的出现极大简化了游戏开发流程，游戏引擎是指用一些已编写好的可编辑电脑游戏系统或者一些交互式实时图像应用程序的核心组件。如今市场上有许多优秀的优秀引擎，每一款游戏都有着自己的特色。比如Unity和Unreal Engine都是非常优秀的引擎。在国内曾经技术匮乏的那个年代，GameBryo和Orge算是最早的两款引擎，GameBryo是用得最多的商业引擎，Orge是用得最多的非商业引擎。

但是现如今，除了少部分大厂在迭代自己的引擎之外，大部分厂商选择了商业引擎。国内厂商更是如此，在3D游戏引擎上的选择都很明确，手机端用Unity，PC端用Unreal Engine。这些引擎不同于以前的游戏引擎，他们的耦合性很低，通用性更好。

1. 需求分析
   1. 游戏整体需求

网页端消灭病毒游戏的实现运行需要几个软件的共同作用，包括网页端的模块设计，游戏主体的模块设计，以及数据库的模块设计。网页端的模块设计又包括前端和后端，网页前端主要是展示给玩游戏的用户的网页界面，前端的代码有些会被隐藏，展示给用户的代码主要有html代码；网页的后端主要是指游戏的后台，服务器等，由游戏的管理者设计，代码由游戏的开发者编写和管理。

* 1. 系统功能需求介绍

需求分析根据总结的系统具体功能需求分析，将本次网页版消灭病毒游戏项目的内容和规则所需分为以下几个系统功能模块

### 2.2.1人物模块需求

游戏需要设置两个不同的角色，角色属性包括一个医护人员和病毒，两个角色进行对抗决出比赛的输赢。并且能够及时判定何时为胜，何时为负。

### 2.2.2地图模块需求

玩家需要在一个固定的地图内竞技，所以游戏地图不能是无限大的，但是游戏地图可以是随机的，所以需要设计一个无限地图生成算法，每局需要生成一次，并且需要保证双方地图一样，才不会使数据缺失。

### 2.2.3炸弹模块需求

炸弹模块需求是为了制定玩家的武器系统，在本款游戏中，玩家可以使用炸弹来作为自己的武器，当炸弹爆炸后，同样具有溅射伤害，这样游戏双方就可以使用炸弹来进行攻击，同样的玩家具有血量系统，一旦玩家的血量被扣除到零，游戏将自动结束。

### 2.2.4道具模块需求

道具主要分为加速、短时间极加速、炸弹威力增加、生命值加一四个道具。用户通过炸毁的泡沫随机生成的道具，对自己的属性进行增强，并决出胜者。

### 2.2.5匹配模块需求

游戏玩家需要通过匹配系统来进行对战，此模块需要保证游戏在网络较差的环境下的流畅性，此模块将会把选择相同难度的玩家匹配到一起并进行对战。

### 2.2.6排行榜模块需求

为了激发玩家的好胜型，排行榜模块也是必不可少的，游戏系统将会把所有玩家的得分信息，和排名信息发布到游戏的排行榜系统上，以此来激励玩家相互竞争。

### 2.2.7治愈币模块需求

游戏的金币系统，玩家可以通过使用治愈币来进行复活等操作，为了防止通货澎涨，游戏治愈币只可以使用充值或者签到方式获得。

### 2.2.8支付模块需求

当治愈币不足时，可以通过支付系统购买治愈币。

* 1. 游戏架构需求分析

本次课程设计主要是以网页形式进行展示的炸弹人之消灭病毒游戏，在网页端需要进行用户之间的实时匹配和实时对战模式，游戏实现需要完成网页端的代码编写，用户登录信息和注册功能的编写，所有游戏用户在数据库中的存储，游戏主体界面和游戏功能的编写以及游戏美术优化的界面设计。

网页前端内容主要是在Java Web上实现的，主要有html、css、ajax、JavaScript等语言实现的，Java Web，是用Java技术来解决相关web互联网领域的技术栈。web包括：web[服务端](https://baike.baidu.com/item/%E6%9C%8D%E5%8A%A1%E7%AB%AF/6492316)和web[客户端](https://baike.baidu.com/item/%E5%AE%A2%E6%88%B7%E7%AB%AF/101081)两部分。主要是用于用户登录游戏网站的网络通信建立，用户实现游戏登陆页面和账户注册页面设计等功能。

网页后端主要包括用户数据库和游戏服务器的配置主要有Java、MySQL自带的语言、jsp等语言实现，数据库是在MySQL进行建立的，MySQL是一个[关系型数据库管理系统](https://baike.baidu.com/item/%E5%85%B3%E7%B3%BB%E5%9E%8B%E6%95%B0%E6%8D%AE%E5%BA%93%E7%AE%A1%E7%90%86%E7%B3%BB%E7%BB%9F/696511" \t "https://baike.baidu.com/item/mySQL/_blank)，用户进行的登录和账号注册信息通过MySQL进行存储和利用，用户的各种游戏属性信息也在数据库中保存。通过前端向后端发送信息以及后端向前端的回复进行网络通信。

本游戏是web端游戏，无需安装操作简洁，仅登录游戏页面即可。

* 1. 相关技术介绍
     1. **cocos游戏引擎**

cocos游戏引擎是触控科技推出的游戏开发引擎，包含了众多功能，包括新建立项，游戏制作，最后到上线打包的工具。可以通过其快速的生成代码，编辑资源，最终可以发布到众多游戏平台。

本次开发使用了cocoscreator工具，版本号是v2.4.3，2015年2月，touch technology正式推出cocos游戏开发一站式解决方案，重新定义无缝工作流程。Cocos集成了Cocos 2d-x、Cocos studio、Cocos code IDE等框架和工具，为开发者提供全套游戏开发解决方案[2]

Cocos基于一个更大的视角，它集成了Cocos家族中的所有开发框架、工具和编辑器。如果cocos2d-x是一个游戏开发引擎，那么Cocos就是一个完整的游戏开发解决方案。它不再只关注“游戏编码”，而是将对开发者的支持延伸到整个工作流程，包括规划、程序、艺术开发的“铁三角”，为手机游戏开发提供专业可靠的一站式解决方案。

* + 1. **mysql数据库**

MySQL 是由瑞典公司 MySQL AB 开发的关系型数据库系统，可以在各种流行的操作系统平台上使用。它具有具有客户端/服务器架构的分布式数据库管理系统。 MySQL完全适用于互联网，用它构建的数据库可以在互联网上的任何地方访问，因此您可以与网络上任何地方的任何人共享数据库。 MySQL具有功能强、使用简单、管理方便、运行速度快、可靠性高、安全保密性强等优点。

MySQL 是用 C 和 C++ 编写的。可工作于多种平台（Unix、Linux、Windows），并提供不同编程语言（C、C++、JAVA等）的API函数；使用核心线程实现多线程很好支持多CPU；提供事务性和非事务性存储机制；基于线程的快速内存分配系统； MySQL 使用双许可，用户可以在 GNU 许可条款下将 MySQL 软件作为免费软件或开源软件使用，或者从 MySQL AB 获得正式的商业许可

* + 1. **Servlet 和JSP 技术**

JSP（Java Server Pages）是Sun指定的一种服务器端动态页面技术的组件规范。 Jsp 是一个后缀为“.jsp”的文件。该文件主要是html和少量java代码。 jsp文件会被转成Servlet在容器中执行。

我的项目中，运行时部分采用了MVVM架构，其实Servlet和JSP也是MVC架构，MVC模式（Model-View-Controller）是软件工程的一种软件框架，系统分为三个基本部分：模型（Model）、视图（View）和控制器（Controller）。

而在这个模型中，Servlet作为Controller而存在，具体过程为首先，web浏览器会发送HTTP请求到服务器，被Servlet获取到后就会开始处理。处理参数并转发等等。

1. 系统设计
   1. 游戏整体架构分析

本游戏采用MVVM架构，共有基础层，领域层，应用层和表现层。MVVM架构其实就是MVC的增强版，该架构正式连接主要的使用原因是模块间的耦合度要小，区分稳定和变更代码。这样在后期维护的时候可以保证为了修改某个功能而不至于修改大量代码，保证结构的稳定。结构图如下。

图示

描述已自动生成

图3.1 MVC架构图

本游戏中，用户登陆登陆界面和游戏界面为分开设计，我称之为“登陆时”和“运行时”状态，两种状态的耦合度极低，当用户在登录后，“运行时”仅会将登陆之后的用户名和金币数量发送给“运行时”，这样就可以保证两种状态只有二元数据的关联。

在“登陆时”，游戏分为经典的前端和后端，前端发送请求给后端，后端通过和数据库交互，再将处理好的数据返回给前端。

在“运行时”，游戏则会变为MVVM架构，游戏所有的UI界面都为表现层，而与之对应的，将会有一个UIViewControler来负责各种UI之间的逻辑。

表现层只有BIND机制，其内部无任何逻辑，只负责绑定数据，将数据发送给UIControler，而在为了实现碰撞体积检测部分时，则无法使用UI控件向UIControler发送数据，此时便可以使用事件系统提出请求，具体实现为将碰撞函数封装为一个事件，这样当每次发生碰撞之后，都会发送一个事件，之后与之绑定的UIControlerCommand就会接受数据并向逻辑层发送数据。

UIControler负责处理表现层绑定的数据和事件，此层的出现也比MVC架构耦合度进一步降低，将UI和逻辑彻底分开。此层允许调用应用层和领域层。但是与领域层的通信并非像与应用层一样可以直接引用，而是需要DataBinding方式来监听数据，也就是说表层只可以监听数据，引用数据，调用其暴露的接口，但反之，应用层和领域层则对表现层无任何的感知，这样就可以保证耦合度大大降低。

应用层的功能主要是数据处理，和提供对外暴露的方法，使其能够处理数据，再通过API和Service进行交互，来获取游戏人物坐标，地图数据，人物状态，等“运行时”的数据。但是应用层本身并不会存储数据，只会处理数据。该层有权利直接调用领域层，也就是model层。但自身并没有数据结构。

领域层为整个程序的数据中心，其内部有完整的数据结构，且内部无任何的方法，只负责存储数据并提供少部分接口用来管理数据，但仅负责取出数据，不会有任何处理数据的行为。数据结构即为2.4中游戏主体变量名所示。Service则为服务类，其主要功能是创建应用层实例，而应用层又会创建领域层实例，Service负责操作 Model，提供模块级别的业务逻辑接口，实现业务逻辑。其还负责与网络进行通信，来实现游戏对战。游戏整体架构如下图所示。

图示

描述已自动生成

图3.1游戏整体架构

* 1. 网络通信架构分析

网络通信架构仅存在于service中，由于采用了P2P机制，所以并没有设置中心服务器，每一个客户端既是服务端又是客户端，所以们使用了房主竞选算法，双方初始化游戏后，均开始进入竞选模式，先进入房间的客户端会将自身上锁，并向对方发送竞选成功报文，对方因为后进入房间，所以无法上锁，当收到对方竞选成功报文后，承认对方是房主，竞选过程完毕。

其主要技术是ajax，其全称是AsynchronousJavaScript+XML，是下面这几种技术的融合：HTML和CSS的基于标准的表示技术，DOM进行动态显示和交互，XML和XSLT进行数据交换和处理，XMLHttpRequest进行异步数据检索，JavaScript将以上技术融合在一起，本游戏中充值页面即采用此技术，用来时刻刷新用户充值状态。由于用户充值后，需要实时更新用户金钱的状态，来时刻保证用户数据保持最新状态。本游戏中充值页面即采用此技术，用来时刻刷新用户充值状态。由于用户充值后，需要实时更新用户金钱的状态，来时刻保证用户数据保持最新状态。由于使用了随机地图机制，为了确保双方用户能保持同步，必须先确定一方为地图生成者，之后再将地图数据发送给另一方。我在初始阶段使用了Cocos Creator中有腾讯云服务，但是腾讯云服务在竞选房主阶段存在问题，所以我自己实现了一个重写的方法，当双方用户进入到房间后，首先向服务器发送一个竞选请求，服务器将先收到请求的一方作为房主。同时向另一方发送竞选结果。竞选成功的一方将成为地图生成者，此过程将使用eval()函数，将生成的地图转换为字符串发送给对方，对方使用eval函数将其变为二维数组，以此来完成地图的解析过程。网络系统模块如图3.2所示。

图示

描述已自动生成

图3.2网络系统架构

* 1. 游戏模块介绍

游戏主要分为登陆时模块和运行时模块，登录时模块又分为登录模块，排行榜模块和注册模块，而这三个模块又会和大后端网页验证，数据库，短信验证码模块相关联。运行时模块分为人物模块，地图模块，道具模块，匹配模块，金币模块。具体模块结构图如图3.3所示。下面将会对这些模块进行具体阐述。

图示

描述已自动生成

图3.3游戏整体模块结构图

* + 1. **数据库模块**

当注册完用户账号信息时，网页会与后台进行联系，数据库连接，前端与后端信息交互，将注册的信息传输到数据库当中并保存下来。

上传的信息不仅包括注册的用户账号、昵称、密码、手机号等信息，还包括用户完成游戏后的治愈币数量、签到日期以及得分等信息，只需要将代码第二行分别改为“date”和“score”就可以实现信息交互。

此次消灭病毒游戏的数据库采用MySQL数据库引擎开发，管理端与用户端共用一个表，游戏用户注册时的信息主要存于该数据库的表中，数据库表的字段名机器数据类型如下表2.5所示。

|  |  |  |
| --- | --- | --- |
| 字段名 | 数据类型 | Not NULL |
| ID | Int | 是 |
| Username | Varchar | 是 |
| Password | Varchar | 是 |
| Score | Varchar | 是 |
| Email | Varchar | 是 |
| re\_coin | Varchar | 是 |
| Phone | Varchar | 是 |
| Day | Varchar | 是 |

表2.5数据库表字段名及其数据类型

ID为玩家游戏ID，此ID用来区分每一位用户。不可重复。Username为玩家游戏昵称，可自定义，可重复。Password为玩家登录游戏账号密码。Score为玩家游戏总得分。Email为玩家邮箱。Phone为玩家手机号码，系统有手机号注册系统，可以通过短信验证码方式来注册游戏账号，也可以通过短信验证码的方式来找回密码。

* + 1. **短信验证码模块**

短信验证码模块具有发送短信，发送短信时间回调，短信验证码存储的功能，该模块主要负责承接注册模块的需求和找回密码模块的需求，当系统需要验证手机号时，会记录手机验证码和对应的手机号，当用户输入手机号后，则会检查二者是否一致。在每次发送完短信之后，都会调用回调，告知前端已经发送完毕。

* + 1. **网页验证码模块**

此模块具有防止用户重复注册注册的功能，防止用户重复发送短信，对其他用户造成骚扰。首先后端会产生一个变形后随机数图片，并生成一串token，并存储在后端，之后再将结果和token发送给后端，后端进行校验后，判断对错，无论对错，后端都会将此字典项删除，防止重复发送验证。

* + 1. **登陆时前端模块**

登录时前端模块包括了登录模块，排行榜模块，注册模块和找回密码模块，他们都使用了三大后端模块，登录模块负责用户登录操作，排行榜模块负责读取数据库并将排行顺序的数值显示给用户，注册模块负责调用短信验证码模块和网页验证码模块来实现注册的功能，找回密码模块调用了短信验证码和网页验证码的功能，实现了用户可以通过手机号发送短信的方式找回密码。

* + 1. **人物模块**

人物模块包括了坐标系统，碰撞系统，武器系统，生命系统。

坐标系统是人物模块的核心，因为本游戏是联机对战游戏，如何保证游戏流程无卡顿的运行，很大一部分难度在于能否处理好坐标问题，因为当用户的网络环境较差时，坐标可能发送的并不连续。而一旦发送的不连续就会导致画面卡顿。所以为了解决此问题，使用了坐标状态转换算法。一旦系统检测到用户网络环境较差，系统将自动从发送坐标改为发送方向向量。发送方向向量的好处是，无需重复发送，只有当用户的方向改变时才会发送，这样就极大的减少了数据传输量。但是此算法一旦遭遇发送的数据丢失就会导致后续数据全部错误，所以此算法需要配合坐标检错算法一同进行。

坐标检测算法的原理很简单，具体实现方式就是每隔五秒钟向对方发送一个坐标信息，并强制对方修改位置信息。这样就既可以保证用户数据的准确，又可以保证数据的发送量不至于过大。

碰撞系统使得用户不至于在移动的时候穿墙而走，将玩家限制在指定的区域内游玩儿。

武器系统中，玩家可以拥有两种武器，第一种武器就是可以往脚下放置炸弹，炸弹爆炸后的碎片如果溅射到自己或者其他玩家就会对其造成伤害，第二种就是发射炸弹，玩家可以通过向对方发射炸弹的方式来攻击对方。

生命系统中，如果某一方玩家的生命值为零，就会自动结束游戏，并宣告胜利方和失败方。生命系统还可以和道具系统进行通信，当玩家获取到指定道具的时候就可以获得额外生命值。

* + 1. **地图模块**

由于游戏匹配是P2P的方式，所以游戏在游玩儿过程中，是不会有中心服务器的，所以必须由一方来生成地图，具体是由谁来生成地图是由匹配模块来决定的，地图模块的主要功能是随机的生成一张10\*10的地图，在生成地图的过程中会与道具模块进行通信，告知道具系统每个道具的位置和数量。地图中有两种砖块，一种可以破坏，而另外一种则无法破坏，可以破坏的砖块被破坏后有一定的几率获取。具体的概率再与道具模块进行同步获得。地图模块中同时会与人物模块进行通信，此时会设置敌我位置，并将此信息通过匹配模块发送给对方。

* + 1. **道具模块**

道具模块主要负责道具的生成，概率的设置。主要与地图模块和人物模块进行数据交换。本游戏包括四种道具，炸弹范围增强，生命值增强，炸弹数量增强，位移速度增强。炸弹范围增强主要控制炸弹的溅射伤害，此操作会和人物模块中的武器系统进行通信，会改变指定玩家的爆炸范围。生命值增强会增加玩家的生命值。炸弹数量增强则会改变用户所能连续放置的最大炸弹数量（包括所能发射的炸弹数量），唯一速度增强则会改变用户每帧所增加的位移量，以此来改变用户的位移速度。

* + 1. **匹配模块**

匹配模块核心思想是P2P匹配，由于客户端在匹配的时候是没有中心服务器的，每一个客户端都是服务器，所以需要一种匹配机制来决定谁是匹配的决定者。在与地图模块进行通信时，需要选出一个房主来生成地图。所以这里采用了房主竞选算法，匹配流程图请看图4.5，核心思想在于开始匹配后，一旦发现在线玩家，就会检查对方的上锁标志位，如果对方未上锁，则向对方发送匹配请求，如果在发送的过程中，对方优先匹配到别的玩家，别的玩家则会先将其上锁标志位锁定，则后匹配的玩家将匹配失败，重新开始搜索在线玩家，一旦搜索成功，并且匹配成功后，则会开始决定谁是房主，一般是发送上锁请求的玩家默认为房主，房主选择成功后，则会与地图模块通信，生成地图，并将生成好的地图发送给另一方，此时，双方完成地图同步。

* + 1. **金币模块**

金币模块主要和充值系统和签到系统进行交互，每次开始游戏都需要花费一定数量的金币，所以金币的作用是决定玩家是否有资格开始一把游戏，玩家可以通过每天签到的方式获取一定数额的金币，也可以通过充值的方式获取金币，当玩家选择通过充值的方式获取金币的时候，金币模块就会和支付模块进行通信，玩家需要扫码进行支付。充值模块具有回调的功能，当充值系统检测到玩家充值成功后，便会告知金币系统，并将玩家的金币数量增加。

3.4**游戏数据结构**

游戏主体开发采用Cocos Creator游戏引擎开发，一些变量名存在于Cocos Ceator游戏引擎内，其他游戏主体全局变量名及其作用如下表2.4所示。

|  |  |
| --- | --- |
| 变量名 | 作用 |
| UserId | 玩家用户名 |
| re\_coin | 治愈币 |
| Speed | 角色移动 |
| Enemy\_Speed | 敌人角色移动速度 |
| Bullet\_Speed | 子弹速度 |
| Bullet\_Max | 子弹上限 |
| Player\_Live\_Num | 玩家生命值 |
| Enemy\_Live\_Num | 对手玩家生命值 |
| Player\_Force\_Num | 无敌标志位，在Dead脚本中使用 |
| Player\_Fire | 玩家放置炸弹 |
| Player\_Towards | 角色朝向 |
| Boom\_Limit | 最多可以连续放几个炸弹 |
| Boom\_speed | 炸弹爆炸速度 |
| Score | 玩家得分 |
| Speed | 玩家移动速度 |
| Is\_host | 判断自己是否是房主 |
| R\_NUM | 可破坏墙体数量 |
| Player\_Boom\_Power | 判断炸弹多久destroy，在power'脚本中使用 |
| TEST | 测试模式 |
| Rec\_Arr\_str | 接收字符串待转化为数组 |

表2.4游戏主体变量名及其作用

1. 详细设计
   1. 登陆时系统设计

登陆时属于经典前端后端结构，前端为html+js+css，用户通过登陆网页来注册账号，当注册账号时，会需要玩家用户填写相关的账号，密码，手机号等信息，填写完成后，根据输入的手机号，向该手机号发送短信验证码，填写验证码，最后填写正确，信息录入数据库，完成注册，如果之前已经注册过该账号，会发送注册失败的相关信息提示用户。注册成功后，即可以登录游戏。游戏网页采用了双重具体流程如图4.1。



4.1.1游戏登录时流程示意图

游戏内，玩家每匹配一次游戏，需要消费两枚治愈币，也就是虚拟的游戏币，每天签到也可以获得10枚治愈币，制定这样的游戏性质，是为了让玩家更有游戏的竞技性，避免游戏没有奖惩机制为玩家带来的无聊感。

本部分使用了AJAX技术，如图4.7，确保了充值过程不会出现错误，当我们提交一个表单请求给服务器，服务器接收到请求之后，返回一个新的页面给浏览器，这种做法浪费了很多带宽，因为我们发送请求之前和获得的新页面两者中很多的html代码是相同的，由于每次用户的交互都需要向服务器发送请求，应用的访问时间取决于服务器的返回时间。

而我们使用Ajax就不同了，Ajax只取回一些必须的数据，它使用SOAP、XML或者支持json 的Web Service接口，我们在客户端利用JavaScript处理来自服务器的响应，这样客户端和服务器之间的数据交互就减少了，然后用户请求就得到了加速。

图形用户界面, 图示

描述已自动生成

图4.1.2AJAX原理示意图

* 1. 运行时系统设计

游戏中设计了存活机制，每隔5秒向对方发送存活报文，另一方收到存活报文后，会返回给发送方一个确认，如果对方5秒之内没有发送存活报文则会终止游戏而重新刷新网页。玩家在开始游戏后，会首先选择难度，只有选择相同难度的玩家才会匹配到一起，这样也保证了游戏的公平性，等待一局游戏结束后，双方会共同决定是否进行下一局游戏，只有双方都选择接受才可以进行下一局游戏。玩家通过游戏所得总分和与全部玩家进行比对，最终排出所有玩家的排行榜。玩家的分数在开始阶段会进行初始化，一旦玩家开始游戏后，每进行完一局游戏都会进行更新，最后完成数据更新，具体过程请看4.2流程图。



图4.2游戏运行时流程示意图

### 4.2.1人物系统

在完成游戏的登录操作后，首先要进入游戏并且匹配游戏玩家，游戏的地图初始界面包括四周的墙壁砖块Block Normal，locker，设计的生两种模块的函数分别问CreatBlock()函数和CreatRBlock()函数，在10\*10的界面内，四周以及地图内，以不同的坐标和位置，生成对应的砖块。

生成游戏本身玩家和生成敌人的代码大概一致，生成敌人的代码还要加上敌人相应的坐标。

### 4.2.2地图系统

在游戏正式开始后，要弹出游戏地图界面，还有匹配到的两个游戏玩家角色。游戏的地图初始界面包括四周的墙壁砖块Block Normal， locker，设计的生两种模块的函数分别问CreatBlock()函数和CreatRBlock()函数，在10\*10的界面内，四周以及地图内，以不同的坐标和位置，生成对应的砖块。

当creat一个初始化地图后，需要游戏双方的地图一致，不仅游戏环境一致，玩家角色出现在的位置，可破坏砖块所隐藏的道具位置和属性也必须一致，这需要游戏双方的确认。生成过程如图4.2.2 。

图示

描述已自动生成

图4.2.2

### 4.2.3道具系统

在炸弹人生成炸弹后，当炸弹范围包括可破坏的砖块时，被破坏的砖块会随机生成具有不同属性的道具，这些道具用来游戏玩家提高自己的游戏属性，包括移动速度增强，炸弹个数增加，生命值增加等。

当玩家放置炸弹后，会对范围内的敌人或可破坏的砖块进行炸毁，受到伤害的角色会减少一条生命，被破坏的砖块会被销毁

放置炸弹后，设计的炸弹伤害方式是，生成几个小元素块，当玩家或可破坏的砖块与小元素块的坐标位置一值时，即为受到伤害，随之砖块销毁，角色生命随之减少。

坐标为正即为向右的伤害，坐标为负即为向左的伤害，同理当后一位为正时，即为向上的伤害，为负时即为向下的伤害。

### 4.2.4匹配系统

在开始游戏时，先选择游戏难度，游戏难度有三个档次，只有选择相同难度的玩家才会匹配到一起。匹配成功后，竞选房主。若双方三秒内均发送存活报文，则房主开始创建随机地图，并将地图发送给加入者。随后双方初始化地图，成功后开始炸弹人游戏。游戏中每0.1秒发送给对方自己的位置。加入一方被游戏炸到则血量减一，血量先为零者输，另一方获胜，游戏结束。然后服务器获取胜利玩家id，并将胜利者分数加10分，将此分数与数据库中最高分进行比较，若该分数大于最高分，则更新数据库最高分，否则不更新。游戏结束后，双方可共同决定是否开始下一局游戏。

4.3坐标系统设计

由于游戏本身为web游戏，所以网络连接性能不如本地游戏稳定，所以就需要一种算法来保证在网络环境较差的时候，仍能保持较稳定的连接状态。所以我设计了帧同步加矢量预测算法，当玩家的网络较差时，系统将自动从发送坐标数据变为发送方向矢量数据，但是如果网络波动较大的话，该算法就会导致发送的数据出现误差，所以该算法又实现了每5秒发送一次坐标来进行校验，通过该方法就保证了发送数据的稳定性。具体流程图如4.4所示。

图示

描述已自动生成

图4.4方向预测算法具体流程

4.5房主竞选系统设计

由于游戏在运行时是没有中心服务器的，所以当玩家匹配成功后就需要从两名匹配好的玩家中竞选出一位房主，房主的职责就是生成随机地图并发送给另一名玩家，这样就可以避免双方同时生成地图而不知道该以谁的地图为准。

房主竞选系统具体细节为当双方开始匹配后，开始匹配的玩家就会遍历当前在线的玩家并且查看是否符合自己的等级，一旦发现匹配成功者，则立即发送给对方匹配成功的讯息，对方收到讯息后则会立即将自身上锁，这样做的目的是，防止对方已经发送了其他匹配信息而导致重复匹配，之后收到讯息的一方会立即向发起请求者返回匹配成功的消息，此时发送信息成功者便成为最终的房主。具体流程如图4.5所示。

图示

描述已自动生成

图4.5匹配流程

4.6 游戏设计难点

本游戏最大的难点在于如何实现同步，由于学生服务器的带宽比较低，所以才用了网络环境检测算法和帧同步方向向量预测算法来提升游戏体验，方向向量预测算法应用了帧同步原理的同时，采用了一种预测算法，其他游戏正常客户端双方会时时刻刻发送坐标信息，而方向预测算法只有当玩家方向向量改变的时候才会发送方向向量信息，同时每隔五秒才会更新一下位置信息，这样当另一台设备没有收到最新的方向向量信息的时候，依然会执行之前的方向向量动作。这样就尽可能的降低了数据的吞吐量，以此来弥补了带宽上的不足。

本游戏同时也实现了多端共同游戏，玩家可以同时在手机上，ipad上，电脑上同时进行游戏，又因为游戏运行时和登录时的耦合度极低，所以一个与不同版本的客户端可以同时进行对战，只要内部数据结构不改变，就可以实现夸版本对战功能。

游戏的主要难点是匹配模块和人物模块，为了保证流畅性，以上两个模块需要同时兼顾流量和精度，在二者之间做一个权衡，这权衡需要算法来把控，也就是网络状态检测与方向向量预测算法，只有二者合理的配置才能够保证系统在网络状态不佳时能够稳定流畅的运行。

1. 测试及运行
   1. 游戏开始界面截图展示

进入游戏开始的界面，有两个选项，一个是开始匹配进行游戏，另一个是技能介绍，对游戏内的技能进行简单介绍，让操作者快速上手。左上角可以显示玩家的姓名，这里玩家的姓名由服务器随机给出了。右上角可以显示玩家自己的分数，数据库最高分以及玩家现有治愈币数量。具体内容可见下图5.1所示。

电脑萤幕的截图

描述已自动生成

图5.1游戏开始界面

进入双人游戏开始的界面。此处是为了模拟联机状态两位玩家匹配的情境。双方可以一起点击开始匹配进行游戏。下一步双方可以进入同一个房间竞选房主。图片左上角有两位玩家各自的昵称。右上角依然是分数，最高分以及玩家双方各自的治愈币数量。具体内容可见下图5.2所示。

电脑游戏的截图

描述已自动生成

图5.1.2游戏双端开始界面

* 1. 技能介绍截图界面展示

进入技能介绍界面，可以看到有四个图标。第一行形似闹钟的图标指的是可以增加玩家移速的道具；第二行形似闪电的图标指的是增加玩家抗体作用范围的道具；第三行带有医疗的十字球形图标指的是增加玩家抗体数，即游戏内炸弹的道具；第四行形似红心的图标指的是增加玩家生命值的道具。具体内容由图5.2所示。

图形用户界面, 文本

描述已自动生成

图5.2游戏技能介绍界面

* 1. 双人联机过程截图界面展示

本游戏双人联机过程中可以选择3个难度档次，分别为新手，中等和高手。玩家可以根据自己对游戏的熟悉程度选择不同难度的联机方式。选择相同难度的两个玩家会随机匹配到一起。具体内容如图5.3、图5.3.1及图5.3.2所示。

图形用户界面

描述已自动生成

图5.3难度选择界面展示

电脑萤幕画面

描述已自动生成

图5.3.1游戏匹配时界面

电脑萤幕画面

描述已自动生成

图5.3.2游戏主界面

* 1. 其中一方掉线截图界面展示

在正常情况下服务器会检查玩家双方是否在线，玩家双方每三秒会向服务器发送存活报文，若服务器没有收到玩家的存活报文，即判断该玩家掉线，另一玩家直接取得胜利。具体内容如图5.4所示。

电脑萤幕画面

描述已自动生成

图5.4其中一方掉线界面展示

* 1. 分出胜负截图界面展示

经过激烈的游戏之后，双方取得胜负，游戏结果展示在屏幕上，游戏结束。双方可以选择是否开始下一局游戏。具体内容如图5.5和图5.5.1所示。

电脑萤幕画面

中度可信度描述已自动生成

图5.5分出胜负界面展示

电脑游戏画面

描述已自动生成

图5.5.1再来一局界面展示

1. 总结

在项目的想法来源于炸弹人游戏以及新冠疫情的影响，本次游戏设计相比以前的设计难度相对较高，需要学习很多方面的知识，将知识结合到一起，才能实现游戏的实现。

在设计项目初期，一款网页端游戏，最简洁可以分为前端和后端，前端主要是网页界面的设计，由html代码完成，包括一些界面的内容设计；网页的后端是网页的后台，比如数据库和浏览器的一些编写，网页填写登录信息时，就是与数据库的信息进行比对，以确认玩家填写的信息无误，网页通过前端向后端发送填写信息，后端为前段发送回复判断是否正确，来完成账号的登录，这也是用户端和管理端的一种联系。游戏设置两个游戏玩家，在一个玩家的网页内生成地图，并将地图发送给对方玩家，并通过比对，进行游戏的实时进行，这也是网页端用户之间的一种网络交互。

但随着逐步深入，还是遇到很多细节问题，例如炸弹人游戏的炸弹不能阻挡游戏玩家的路线。炸弹能否对玩家自己造成伤害等等一些逻辑；对于游戏，也有对玩家游戏体验舒适度的考虑，比如地图的大小是否能让玩家舒服的玩好一局游戏，既让玩家体验到游戏时间长度，又不能让玩家的一局游戏持续太长时间，这都需要调试不同的数据来完成测试。游戏设计的平台有所了解后，又一项难处是如何在网页端实现游戏的进行，保证在同一局域网内的玩家能够实时进行游戏，这需要我排队网页的相关知识有一定的学习。

在不断克服困难的过程中，也获取了很多以前没有的知识，让人受益匪浅。

参考文献

1. [美国] 尼古拉斯·泽卡斯. JavaScript高级程序设计[M]. 北京: 人民邮电出版社
2. 冯乐乐Unity Shader入门精要[M]. 北京: 人民邮电出版社
3. 满硕泉Cocos Creator 完全使用手册[M]. 北京: 机械工业出版社
4. [美]Robert Nystrom游戏编程模式[M]. 北京: 人民邮电出版社
5. [美]福勒重构：改善既有代码的设计 [M]. 北京: 人民邮电出版社

[6] Ted Husted. STRUTS IN ACTION[M]. 北京: 机械工业出版社, 2005.  
[7] Patrick Grassle Henrieete baumann Philippe Baumann,UML2.0实战—项目开始指南[M]. 北  
京: 人民邮电出版社, 2007  
[8] 张兴科. JSP动态网站设计项目教程[M]. 北京: 中国人民大学出版社,2010.  
[9] 张全新, 李炜译. 数据库驱动的Web站点[M ]. 北京: 清华大学出版社, 2002.  
[10] 于大伟. 基于WEB的数据库应用系统优化管理解决方法[D]. 吉林大学, 2005.  
[11] 赵增敏. JSP网站开发详解[M]. 北京: 电子工业出版社  
[12] Shari Lavrence Pfleeger. Software Engineering[M]. 北京: 高等教育社, 2003  
[13] 邹劲松. 物业管理信息系统[M]. 北京: 机械工业出版社, 2004.  
[14] Roger S.Pressman, 郑人杰, 马素霞等. 软件工程:实践者的研究方法(原书第7版)[M]. 北  
京: 机械工业出版社, 2011  
[15] 王珊,萨师煊. 数据库系统概论[M]. 高等教育出版社,2006.  
[16] Abraham Silberschatz, Henry F.Korth, S.Sudarshan. Database System concepts[M]. USA:  
The McGraw-Hill Companies,Inc.  
[17] James Rumbaugh, Ivar Jacobson, Grady Booch. The Unified Modeling Language Reference  
Manual[M]. 北京: 机械工业出版社, 2006.1  
[18] 罗国庆, 陈良萍. 网站建设案例精粹[M]. 北京: 电子工业出版社, 2004.  
[19] Abraham Silberschatz. Database System Concepts[M]. 北京: 机械工业出版社, 2003.  
[20] 朱爱红, 余冬梅, 张聚礼. 基于B/S软件体系结构的研究[J]. 计算机工程与设计, 2005, 26(5):1164-1168.  
[21] Christine Hofmeister. Applied Software Architecture[M]. 北京:电子工业出版社, 2003.  
[22] 李刚. 轻量级Java EE企业应用实战(第3版):Struts 2+Spring 3+Hibernate整合开发[M].

致谢

转眼间，四年大学时光匆匆流过，时光如白驹过隙，感谢刘芳老师这些年来的关照，帮助，让我能在大学四年中学到更多的东西。

感谢我的同学们，让我学习到了很多课本上所学习不到的知识，让我将计算机知识变成自己的兴趣，而不仅仅是去学习。

感谢我的导员，在生活中照顾着我，在百忙之中关照着大家，让我们在学习之余的大学生活中更加舒适。

感谢大家这么多年来的悉心照顾，这四年来我收获了远远比知识更重要的东西，那就是兴趣，有了兴趣的加持，让我感觉我的学习专业知识并非是一件枯燥和乏味的事情，让我对游戏开发产生了很强的动力。

最后我要感谢自己这么多年来的努力，大家的帮助固然重要，但是如果自己不努力，也只是徒劳而已，所以最后我要感谢自己这么多年来的付出，最后祝大家，祝老师，祝自己前程似锦。