<火树银花星满昼>

软件架构文档

版本 <2.0>

修订历史记录

|  |  |  |  |
| --- | --- | --- | --- |
| **日期** | **版本** | **说明** | **作者** |
| <2021日/7月/23年> | <1.0> | <第一次修订架构文档> | <林舒怀，康艺潇，林祺龙，丁逸舟> |
| <2021日/9月/10年> | <2.0> | <第二次修订架构文档> | <林舒怀，康艺潇，林祺龙，丁逸舟> |
|  |  |  |  |
|  |  |  |  |

目录

1. 简介 4

1.1 目的 4

1.2 参考资料 4

2. 用例视图 4

3. 逻辑视图 4

3.1 概述 4

3.2 在构架方面具有重要意义的设计包 4

4. 进程视图 4

5. 部署视图 4

6. 实现视图 5

7. 技术视图 5

8. 数据视图（可选） 5

9. 核心算法设计（可选） 5

10. 质量属性的设计 5

软件架构文档

# 简介

## 目的

本文档将从构架方面对系统进行综合概述，其中会使用多种不同的构架视图来描述系统的各个方面。它用于记录并表述已对系统的构架方面作出的重要决策。

## 参考资料

[1]沈备军. 软件工程原理. Print. 高等学校软件工程系列教材.

[2]周世明,燕飞.AR技术在教育领域的应用研究[J].新闻研究导刊,2021,12(02):61-62.

[3]杨子力.基于AR技术的涉电公共安全宣传教育应用研究[J].云南水力发电,2020,36(08):244-247.

[4]王罗那.增强现实技术（AR）在数学教育中的应用现状述评与展望[J].数学教育学报,2020,29(05):91-97.

[5]袁丽平.基于5G的VR/AR在高等教育教学的应用[J].现代信息科技,2020,4(18):190-192.

[6]高政霞.AR虚实融合推动教育改革[J].科技风,2020(26):99-100.

[7]栾婷婷,董楠楠,刘梅芷.基于移动设备的AR技术在社区教育的应用研究[J].信息通信,2020(08):272-273.

[8]黄任勇.AR增强现实技术及其在教育中的应用研究[J].动漫研究,2020(00):188-190.

[9]李成.5G视域下云计算在移动学习中的应用研究[J].中国教育技术装备,2020(10):49-51.

[10]梁成亮.增强现实(AR)技术下的计算思维教育游戏设计[J].文化产业,2020(14):19-20.

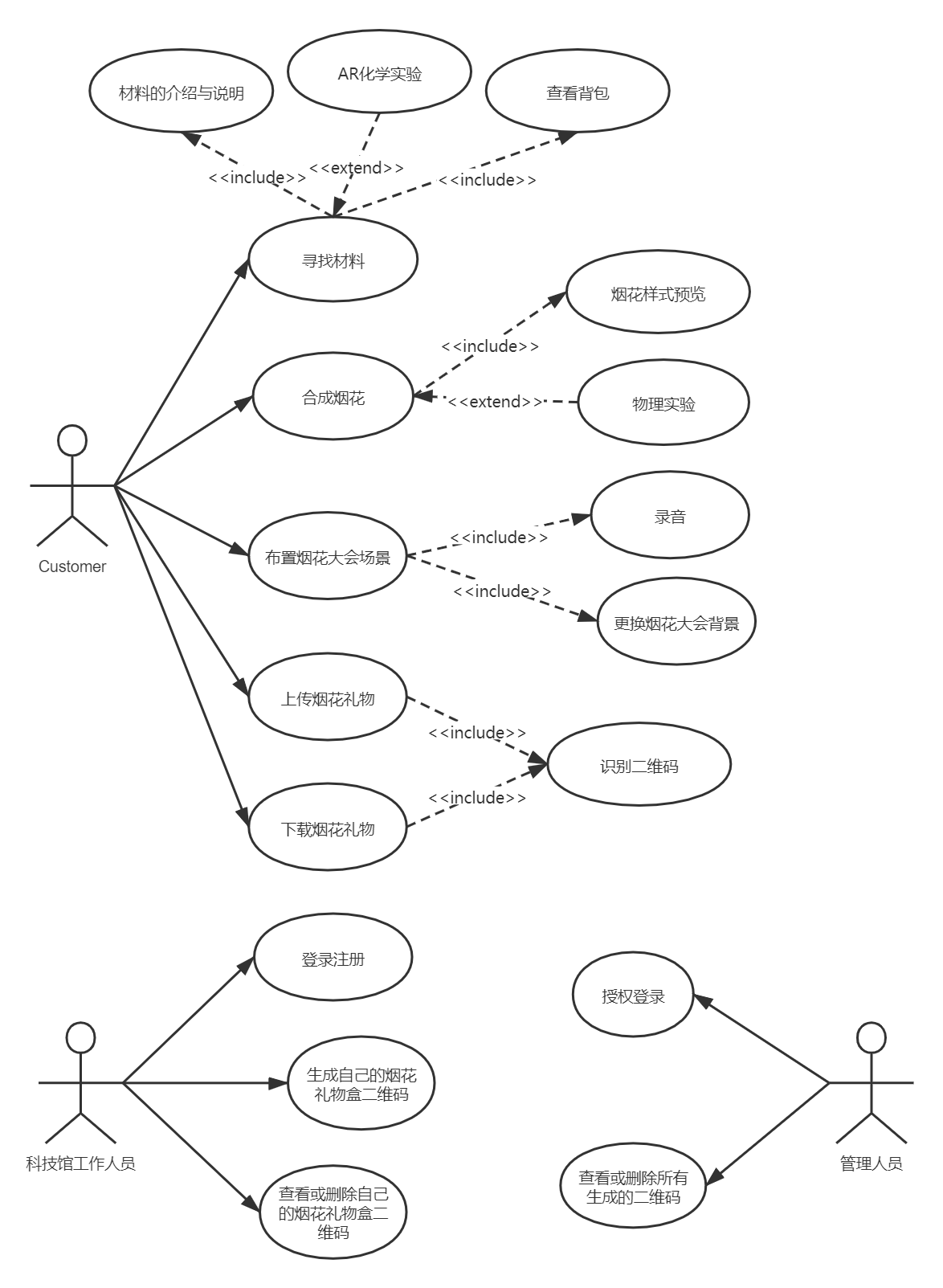
[11]Serkan Solmaz,Tom Van Gerven. Automated integration of extract-based CFD results with AR/VR in engineering education for practitioners[J]. Multimedia Tools and Applications,2021(prepublish).

[12]Lin HaoChiang Koong,Lin YuHsuan,Wang TaoHua,Su LunKe,Huang YuehMin. Effects of Incorporating AR into a Board Game on Learning Outcomes and Emotions in Health Education[J]. Electronics,2020,9(11).

[13]Jorge F. Figueroa-Flores,Lisa Huffman. Integrating AR and VR in Teacher Education: What Pre-service Teachers Perceive[J]. Frontiers in Education Technology,2020,3(4).

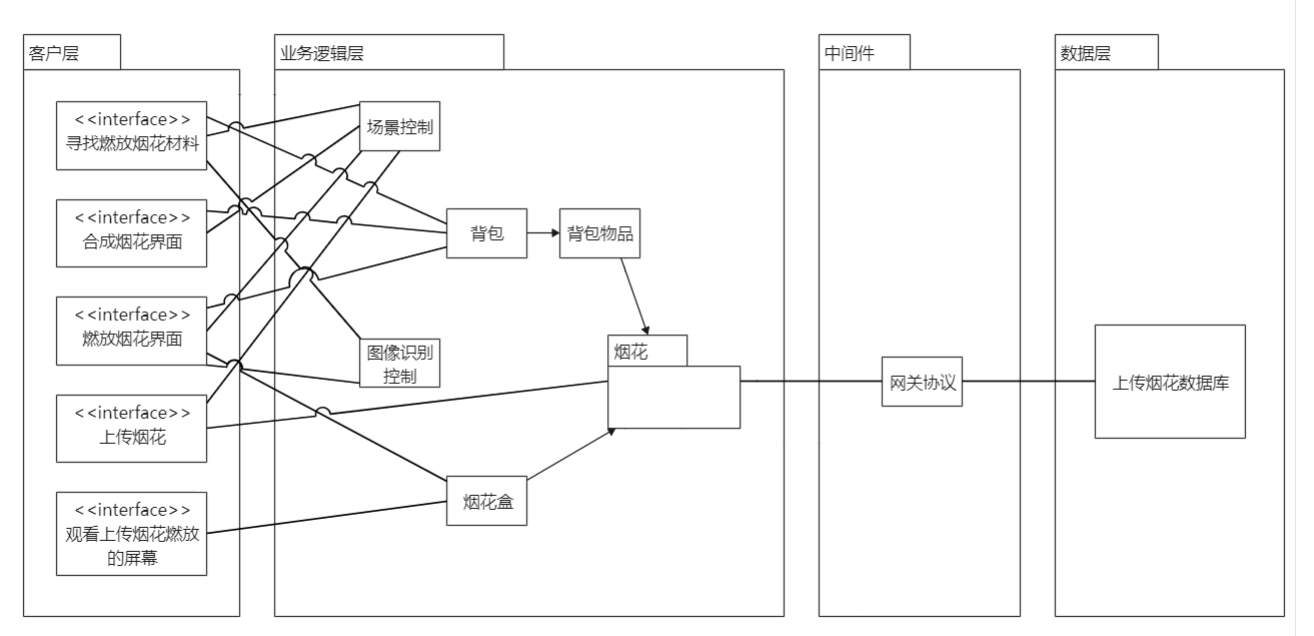
[14]Hasan Köse,Nevin Güner-Yildiz. Augmented reality (AR) as a learning material in special needs education[J]. Education and Information Technologies,2020(prepublish).

# 用例视图



# 逻辑视图

## 概述



软件架构采用三层架构，分为客户层、应用层（业务逻辑层与中间件）以及数据层。

### 客户层

客户层包含了所有展示在用户面前的界面。包含寻找材料、合成烟花、燃放烟花等界面。

### 业务逻辑层

业务逻辑层包含场景控制类、背包类、背包物品类、图像识别控制类、烟花盒类、烟花类等。业务逻辑层主要用于接受用户操作的信息，对用户操作做出处理，响应用户的操作并在必要时与数据库进行交互。

### 中间件

中间件包含了网关协议，还包括图中未标识的数据库开发库等。起到连接应用层和模型层的作用，并在网络传递信息等操作中调用外部结构。

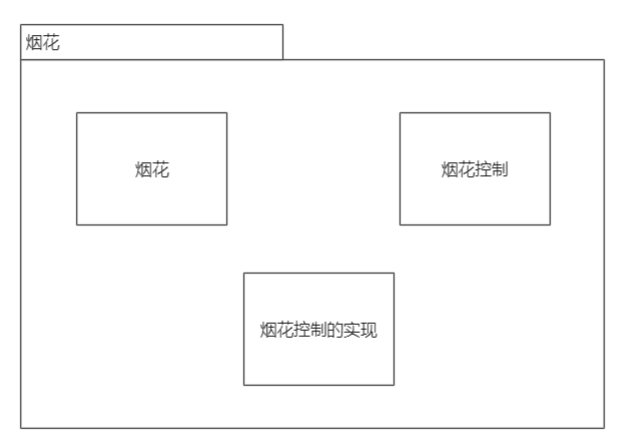
### 数据层

数据层目前包含一种数据库，用于储存上传烟花的信息。

## 在构架方面具有重要意义的设计包

### 烟花包

这个包主要由记录、储存烟花的相关信息的烟花类，控制烟花燃放的烟花控制类，烟花控制的实现类构成，主要用于处理烟花部分相关的逻辑。



#### 烟花类

主要用于记录烟花中各种信息的类，如烟花的初速度、烟花的颜色、烟花燃放的角度等。

#### 烟花控制类

主要用于控制烟花的燃放过程，如燃放位置、开始燃放、停止燃放等。

#### 烟花控制的实现类

主要用于实现烟花控制类中规定的各类函数，是一个实现类。

# 进程视图

## 概述

图中总共有3个进程。其中，请求产生进程运行在客户的手机端与科技馆的电脑端上，负责提交用户自定义的烟花场景与请求烟花场景数据。业务逻辑控制进程负责执行Web层各种任务，用于接受客户端请求、执行业务逻辑、并将响应发回给客户端。数据库管理系统进程负责管理存储在数据库中的烟花数据。

# 部署视图

## 概述

软件分为四个节点，分别为游客客户端、科技馆客户端、服务器、数据库。其中请求产生进程对应游客客户端与科技馆客户端，业务逻辑控制进程对应服务器节点，数据库管理系统进程对应数据库节点。

## 游客客户端

游客客户端运行环境为游客自带的手机，可以通过科技馆提供的无线网络与服务器连接。游客客户端可以让用户自定义烟花，并上传烟花数据。

## 科技馆客户端

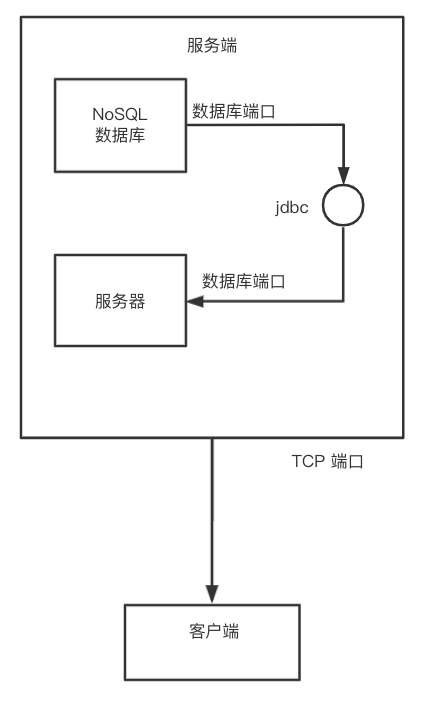
科技馆客户端运行环境为科技馆提供的电脑，通过科技馆提供的无线网络与服务器连接。科技馆客户端从服务器获取用户上传的自定义烟花数据，并进行展示。

## 服务器

服务器将用户上传的烟花数据存储到数据库中，并根据上传时间等参数决定发送给科技馆客户端进行展示的烟花。

## 数据库

# 实现视图



# 技术视图

编程语言： C#

开发工具： Unity

框架： MVC框架

数据库： Neo4j, MySQL

中间件： 游戏开发功能的程序库/引擎子系统 ：粒子系统VFX，Unity 物理引擎

# 数据视图（可选）

MySQL关系型数据库存储烟花的参数：

|  |  |
| --- | --- |
| Firework | |
| PK | firework\_id |
|  | designer\_name |
| type |
| color |
| speed |
| angle |
| radius |

|  |
| --- |
| Firework |
| - firework\_id:Integer  - designer\_name:String  - type:Integer  - colors:list<String> |
| + getFireworkId();  + getDesignerName();  + setDesignerName ();  + getType();  + setType();  + getColors();  + setColors(); |

MongoDB非关系型数据库存储图片

# 核心算法设计（可选）

# 质量属性的设计

易用性：由于该项目致力于AR教育，对于用户来说。基础的使用操作培训时间应该越短越简单越佳（包括ARbook的阅读以及简单的实验操作，如对化学实验的简单使用）。项目采取B/S和C/S混合模式，其中C/S架构的C端有非常强的处理能力，所以在交互表现和安全方面比较突出。具体实现有在部分场景有对应的指南按钮，可以查看相关功能的使用方法，帮助使用者进行相关操作，或者指引使用者进行系列学习。用户可以上传图片到服务器，即时看到图片粒子化的烟花效果。

可靠性：可用时间百分比应达到99.99%，对于突发故障，修复时间平均应该在四小时左右，而最长的故障修复时间应当不超过一天，以免很大程度影响使用者效率，项目采取B/S和C/S混合模式，其中B/S浏览器的方式使得用户的使用十分方便，用户可以何时何地通过Internet访问URL而进行相应的工作，项目故障修复、升级维护也能比较集中。

适应性：本系统采用B/S和C/S混合模式，支持脱机方式，因此能够保证用户随时随地访问系统。同时，系统采取容错技术，具备数据恢复功能，能够保证用户随时随地操作系统。

性能：响应时间方面，所有操作的平均响应时间应当为1秒，对于用户的一般响应时间为0.5秒以内。系统最少应当可支持200人在线使用软件，并保证性能不受影响。我们选取的NoSQL数据库为MongoDB，其中关于内存管理的代码非常精简，有操作系统进行托管。 MongoDB 服务器使用的虚拟内存也非常巨大，并将超过整个数据文件的大小。在《Mongodb亿级数据量的性能测试》一文中，MongoDB 展现了强劲的大数据处理性能。另外，在项目后期会对响应时间和并发性能进行测试以保证项目产品的性能。