# Partake模块详细设计

## 概述

模块需求：该模块负责保证服务器扩展、服务器正常运行和服务器负载分担。用户访问服务器时需要先通过服务器负载均衡模块，负载均衡模块同时负责监测各个服务器正常运行。

涉及需求：

R1服务器高可靠性 C1 服务器故障应该在10min内侦测出来

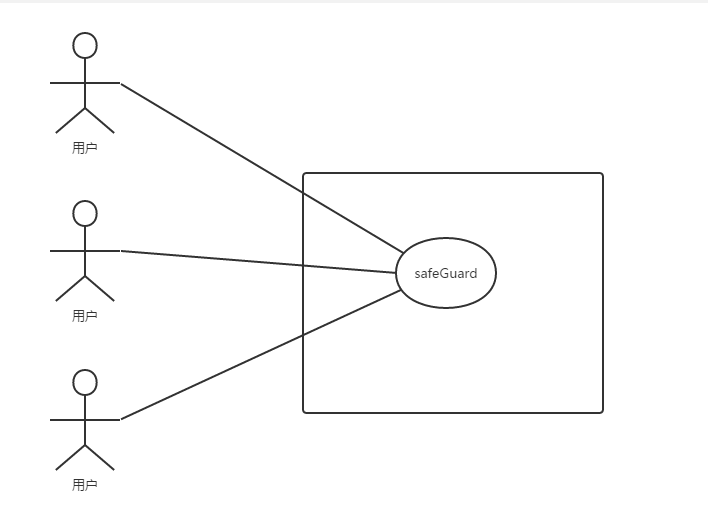
C2 服务器故障时不能影响正常访问

R6 可维护性 C10 允许在系统运行状态下更新系统组件,系统信息

R7 系统运行容故障性 C11 系统存储的数据要每3小时保存一次

R23容故障性 8 C29 对系统存储的数据进行备份

C30 每隔30秒定时保存



说明：用户访问服务器时需要先通过服务器负载均衡模块

**可能会发生的修改实现**

增加服务器

减少服务器

**分配职责**

1. 该模块负责服务器的扩展、负载均衡和保证服务器故障时正常运行

**对外接口：**

接口：void doVisit(HttpRequest request);

功能：当用户的请求到达系统并且通过安全检查后，调用该接口，该接口将选择工作正常并且负载较低的服务器响应请求

**参数说明：**

HttpRequest request：用户请求信息，含有用户的IP、访问时间、mac地址、访问目标等

**返回值说明：**

无

**使用方法：**

每当一个用户向服务器请求数据的时候，调用该模块以选取服务器响应用户请求

**前置条件：**

负载均衡服务器工作正常

**后置条件**

无

**设计方案分析**

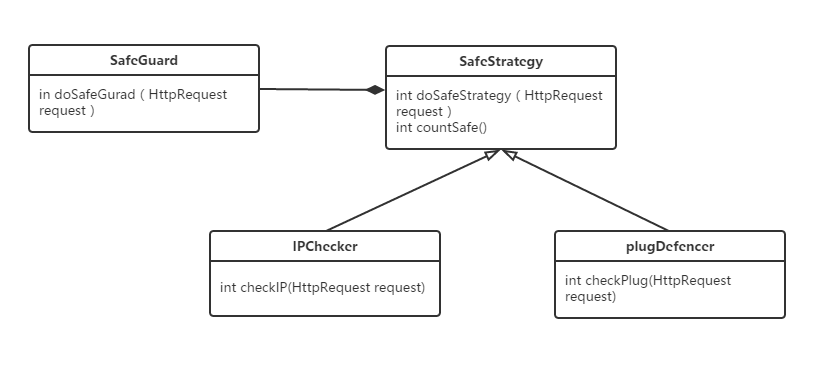
负载均衡模块使用单例模式以保证只有一个实例

负载均衡模块定时对所有服务器进行心跳测试以确保所有服务器正确运行

## 2类图

**静态结构**

**详细设计类图和类图说明：**



设计说明：该模块主要由两个部分组成：SafeGuard 和SafeStrategy

HttpRequest通过SafeGuard类传入SafeStrategy类，再通过SafeStrategy类调用其子类，最后通过子类返回值计算出该次request的安全评级并返回

## 3类描述

**类方法（重要方法要给出前置与后置条件）与重要数据结构描述**

|  |  |
| --- | --- |
| 类 | 说明 |
| Partake | 传入HttpRequest对象，调用负载最低的服务器响应请求 |
| CilentChecker | 检查各个服务器是否正常，并返回该服务器的负载 |
| CilentFeedback | 服务器用于反馈自己的负载 |
| CilentBackup | 策略类的子类，检查IP是否在屏蔽列表内 |

|  |  |
| --- | --- |
| Partake | |
| doVisit | 获取HttpRequest，调用负载最低的服务器响应请求 |

|  |  |
| --- | --- |
| CilentChecker | |
| CheckStart | 开始执行服务器监测 |
| GetFreeClient | 获取最空闲的服务器 |

|  |  |
| --- | --- |
| CilentFeedback | |
| Feedback | 返回自身的负载 |

|  |  |
| --- | --- |
| CilentBackup | |
| Backup | 保存服务器以及数据库数据到备份服务器 |
| Restore | 从备份服务器服务器以及数据库数据 |

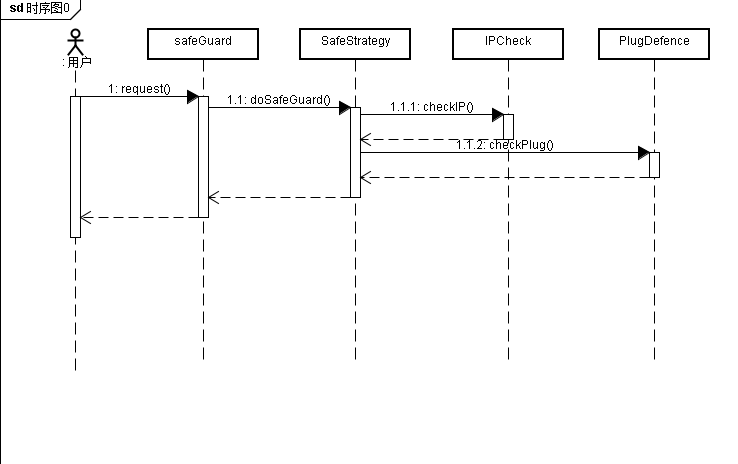
**重要的数据结构描述**

用户请求

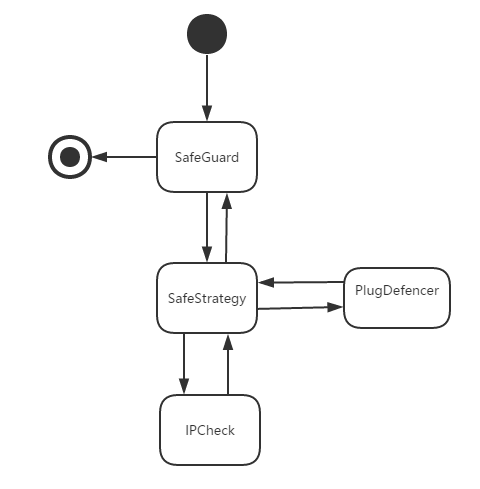
HttpRequest：其中有用户IP、mac地址、访问时间、访问目标

## 4重要协作

**顺序图**



**状态图**



## 5使用的设计模式

**单例模式：**partake类采用单例模式，将构造函数隐藏并提供公共的getInstance方法获取该类的对象，以保证partake类只有一个实例

**场景：**调用partake模块相关方法

**目的：**保证partake类不会拥有多个实例