

**2022年秋季学期**

**计算机系统安全**

**Lab 1实验报告**

|  |  |
| --- | --- |
| 姓名 | 吴昊，袁野，王泽鑫，刘昕烨 |
| 学号 | 120L020818 |
| 班号 | 2003201 |
| 组长 | 吴昊 |
| 组长电子邮件 |  |
| 组长手机号码 |  |

目录

[一．实验内容 3](#_Toc17642)

[二．实验过程 4](#_Toc30359)

[2.1 系统页面介绍 4](#_Toc361)

[2.2 实验步骤 6](#_Toc4889)

[2.2.1 用户身份管理： 6](#_Toc21126)

[2.2.2 游戏列表管理： 7](#_Toc22003)

[2.2.3 游戏运行检测： 8](#_Toc26876)

[2.2.4 游戏运行检测： 10](#_Toc28836)

[2.2.5 日志上报接口 11](#_Toc3406)

[2.2.6 其他问题 11](#_Toc31388)

[三． 分工 12](#_Toc3651)

[四． 总结 12](#_Toc4171)

# 一．实验内容

单机系统上的管理目标：在linux系统上设计一款青少年游戏时长监测管理系统。家长利用该系统能够管理孩子的游戏时间，进而落实国家对青少年玩游戏的管控规定，为青少年的健康发展保驾护航。

编制实验报告，给出系统安全策略文档、系统设计报告，源代码、实现过程说明，及心得体会。

**应用需求1**：管理linux系统上安装的游戏（一个或多个）权限。注册游戏账号时需要登记用户出生年月（可选项：提供人脸照片，以做青少年身份判定），以分析用户年龄。监测系统提供出生年月的验证接口。（可选项：模拟出生年月的验证；）

**应用需求2**：为家长提供了解孩子玩游戏状况的查看界面，以日为单位、周为单位，特殊时段（如深夜）玩游戏的状况。游戏运行时需记录玩游戏的身份、时间、时长，并进行统计分析。（40分）

**应用需求3**：报警和停止游戏功能：对玩游戏时长超过一定时间（可配置，如1个小时）后给出警告；玩游戏时间超过红色警戒线，直接停止游戏，并且当天不能再玩游戏。（25分）

**应用需求4**：检查是否绕开身份检测追踪：检查青少年是否以父母身份玩游戏、是否绕开人脸识别等（如不能实现，列出绕开追踪的方法和可能的检测方法即可）。（15分）

**应用需求5**：为游戏监管部门提供上报接口，以便政府等部门分析评价青少年玩游戏的管理效果。（5分）

**项目实现要求：**

1.综合运用第3章基础知识，理解用户、组、文件权限管理方法、root何时设置setuid、进程运行过程中用户的切换等知识，设计实现青少年游戏时长的监管系统。监管系统的功能不限于上面的应用需求，可自行扩展，扩展部分为加分项。

2.3-4人一组，共同完成系统设计、实现。每人完成一部分，实验报告中能够区分出不同人对系统的贡献。

3.完成系统，系统功能相对完整，并提交系统设计和实现文档。

4.各部分按完成度给分：完成基本功能，给60%分数；设计思路较好、代码规范可读性好，完成质量良，给80%；功能完整、代码完成优秀，给100%。

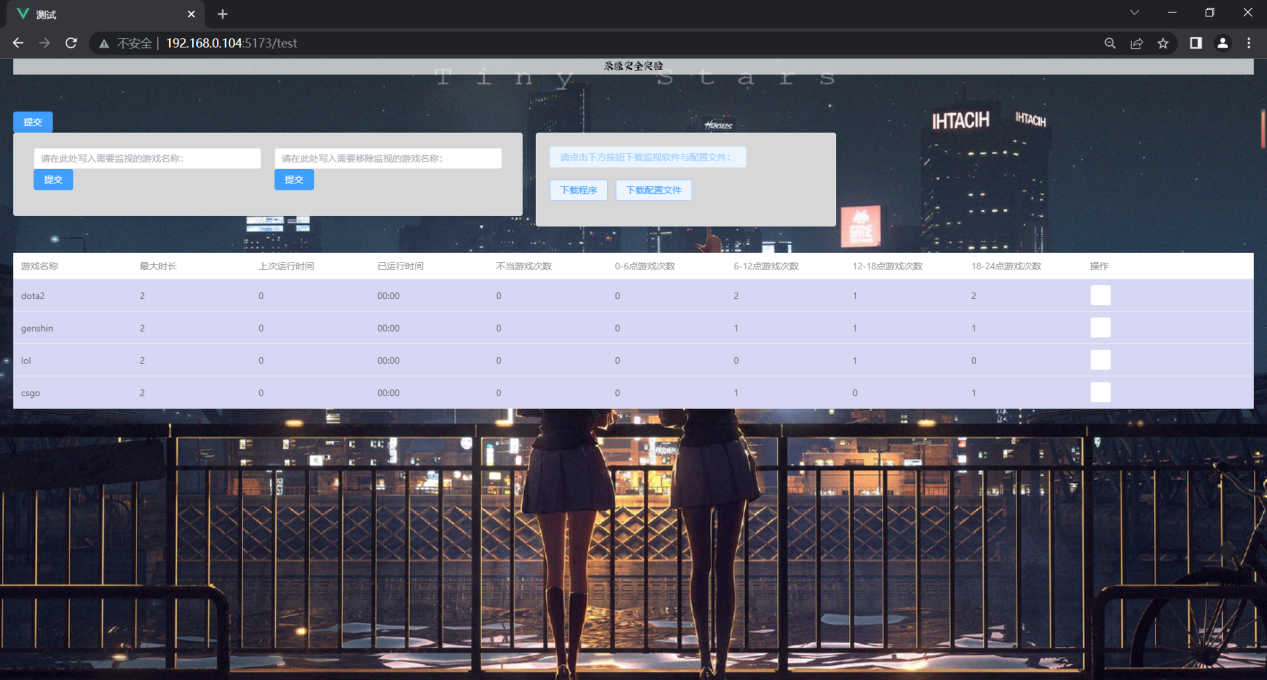
# 二．实验过程

本实验设计的应用场景是Ubuntu系统下的多用户进程管理与归档。

## 2.1 系统页面介绍

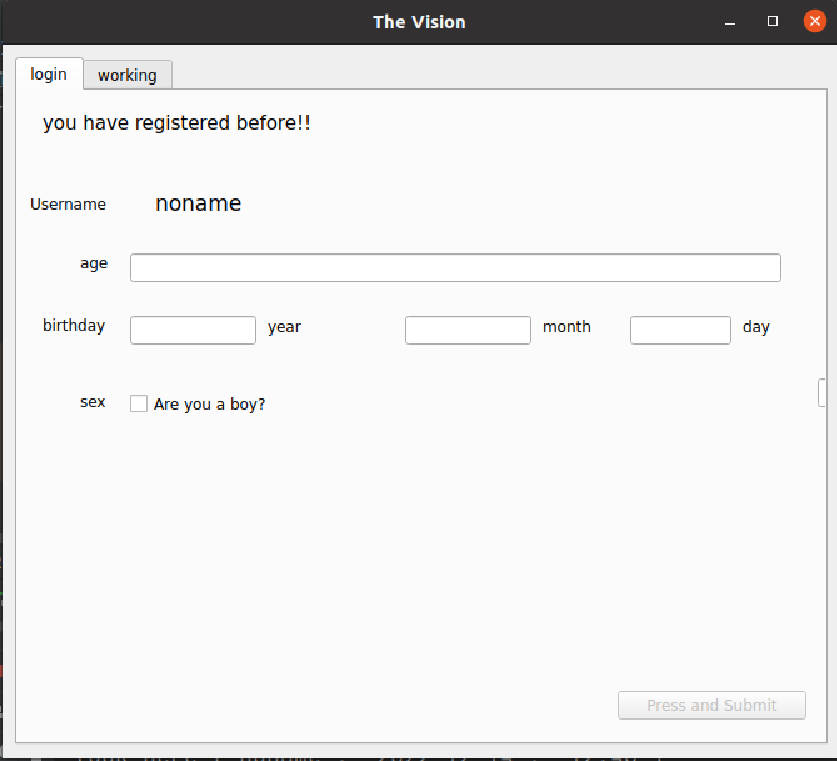
1.建一个服务网站。跟踪常用游戏网站，作为青少年玩游戏的检测范围，列表可更新。用户使用后可向服务网站提供青少年玩游戏状况的日志信息，用以汇总、统计各年龄段、各时间段的玩游戏状况。评价各游戏公司对青少年玩游戏的管理状况。

服务网站外观如下，其可以实现显示要监控的游戏管理列表功能。同时，家长或管理员可以在此页面动态地增删游戏，即需要监视的游戏列表可动态加减。

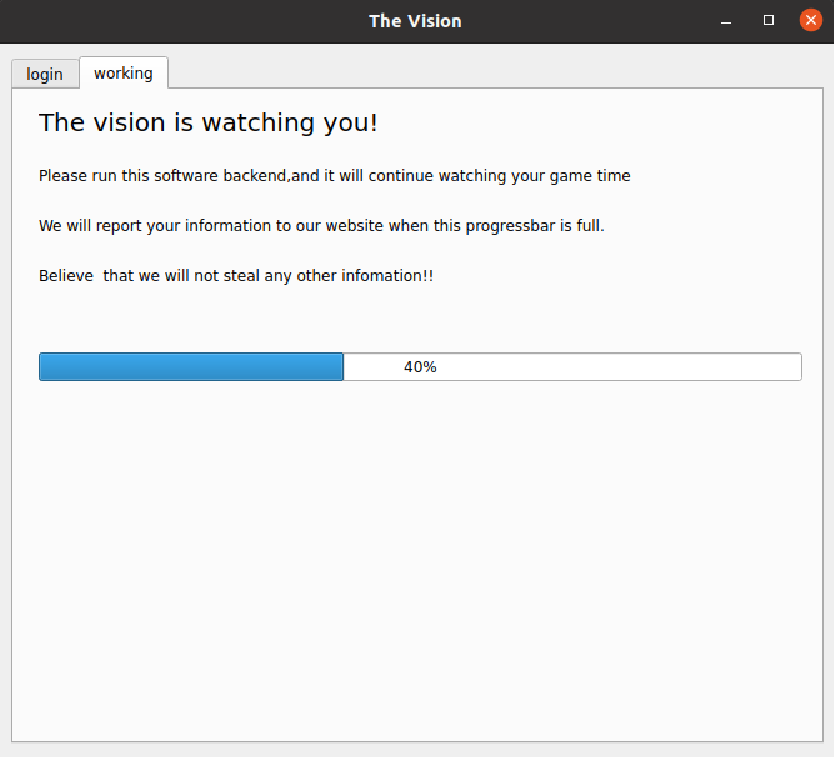


2.在上述页面提供了客户端可下载的单机版游戏管理程序，该程序使用python的pyqt5库开发，外观如下。

该应用分为两个页面，可以相互切换，其一为注册页面，该页面可以登记用户身份界面和接口，并根据用户填写的信息生成初始的配置文件。另外，用户所填写的年龄，也会随年份更新管理。由于我们使用的用户名和Ubuntu的用户名相同，所以可以根据Ubuntu的用户名生成不同的身份配置文件，自然地进行身份更改管理。



当我们填写玩注册信息并点击右下方的按钮之后，我们可以进入第二个页面，所示如下。在这个页面中，填写了一些提示信息，并且其中有一个进度条，当进度条满了之后便会向后台程序发送本机程序进程的信息，以做到随时监控进程的执行。

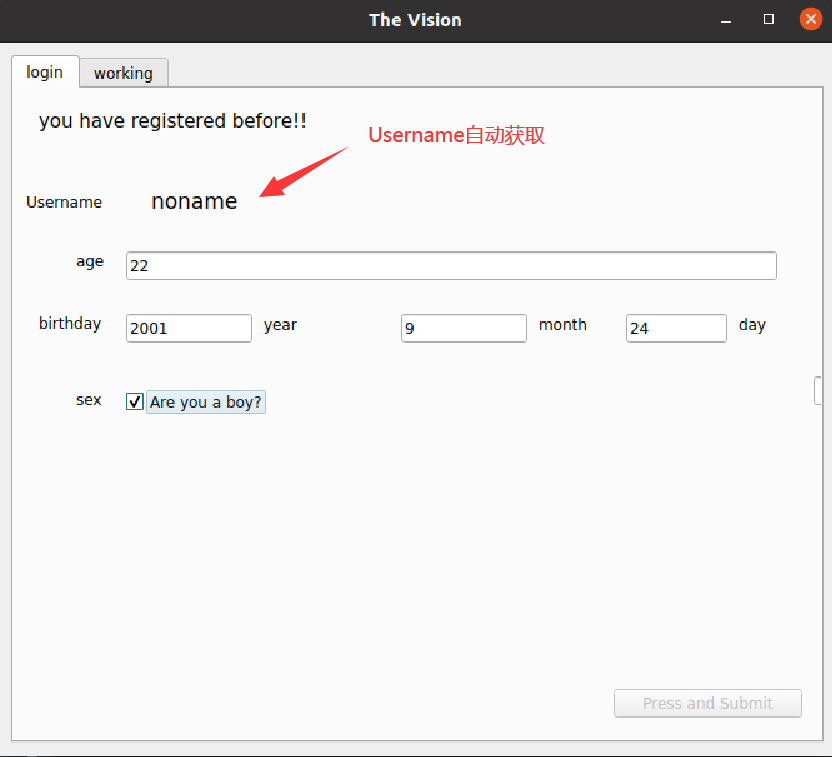


## 2.2 实验步骤

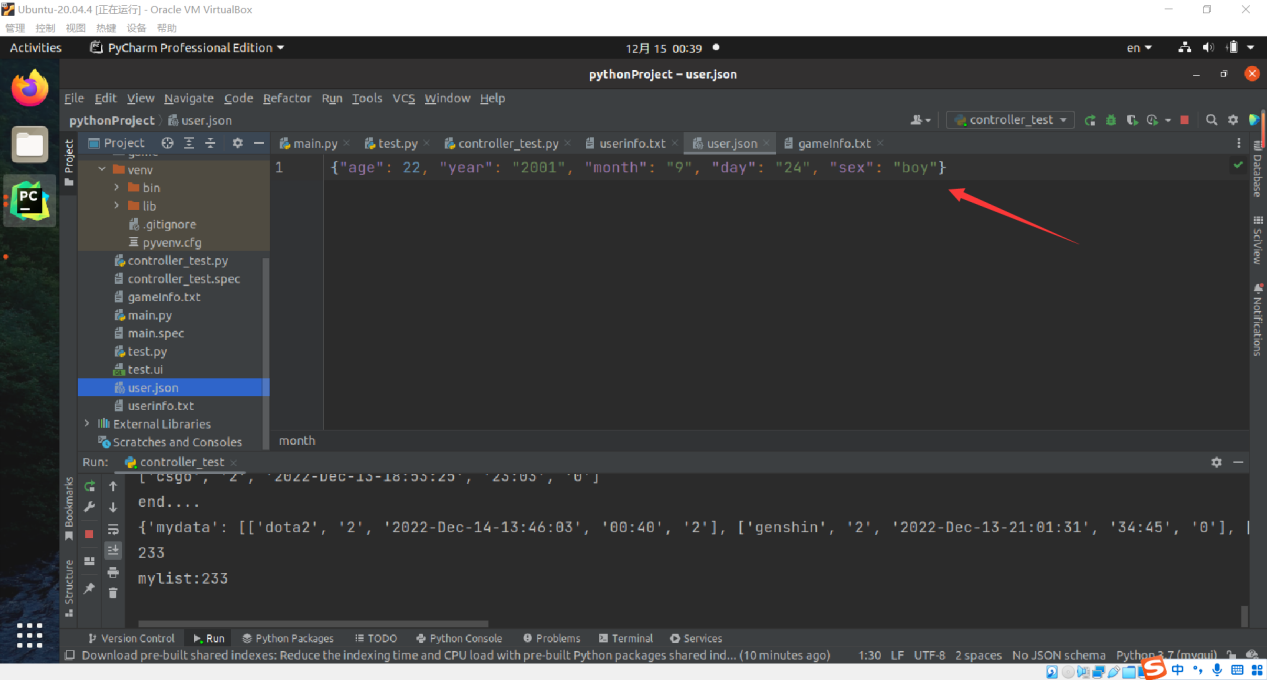
本部分根据实验1原文档与后续的实验1说明文档进行顺序实验步骤的阐述。

### 2.2.1 用户身份管理：

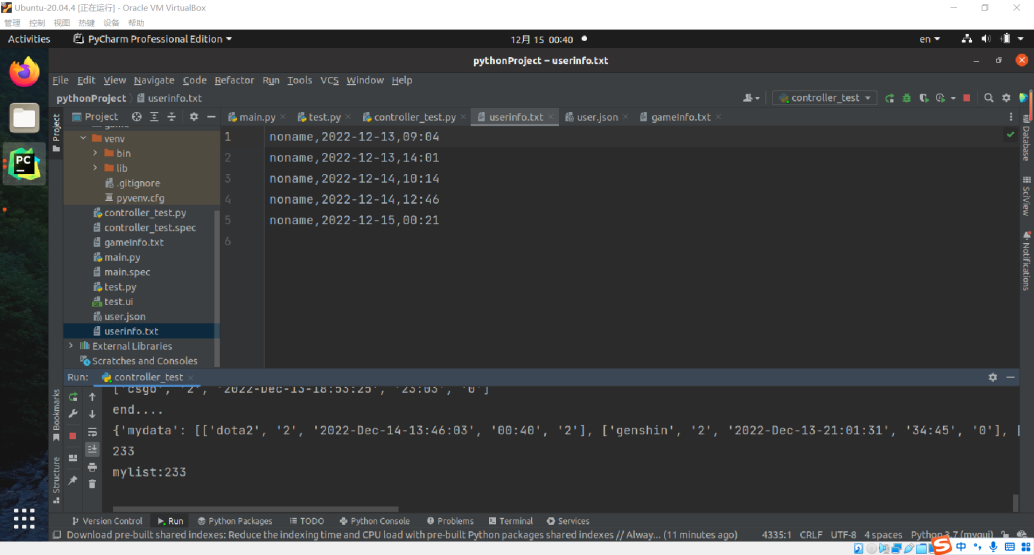
首先是录入界面，根据上文所述我们可以在客户端程序中进行记录。



点击该按钮后，就可以生成配置文件user.json，显示如下：

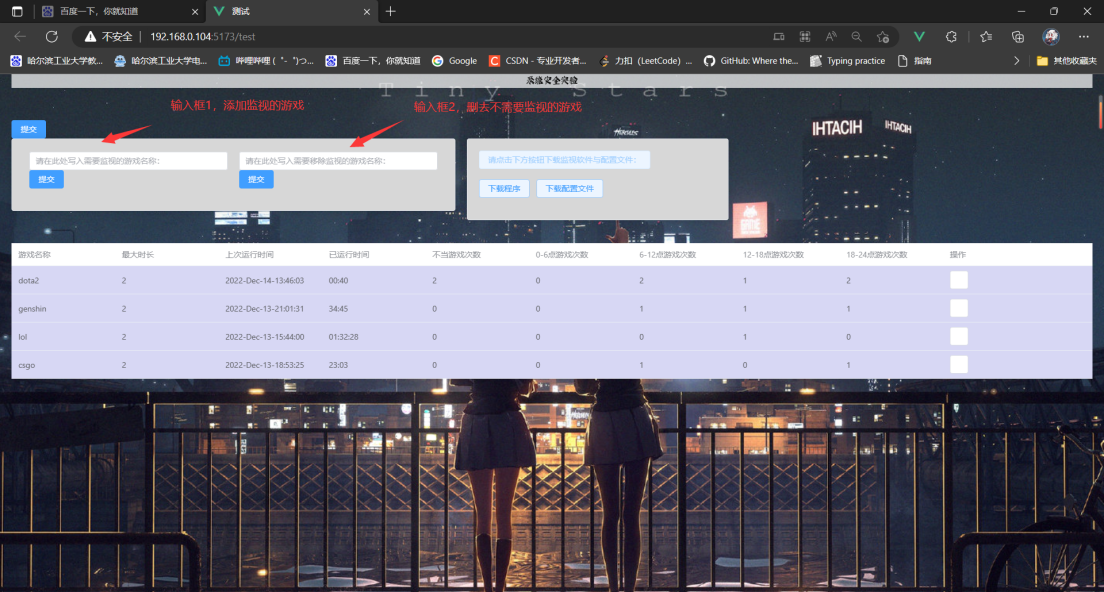


之后，当我们用户登陆系统，打开该软件的时候，我们都会记录该用户的登陆时间，记录在userinfo.txt中，显示如下：

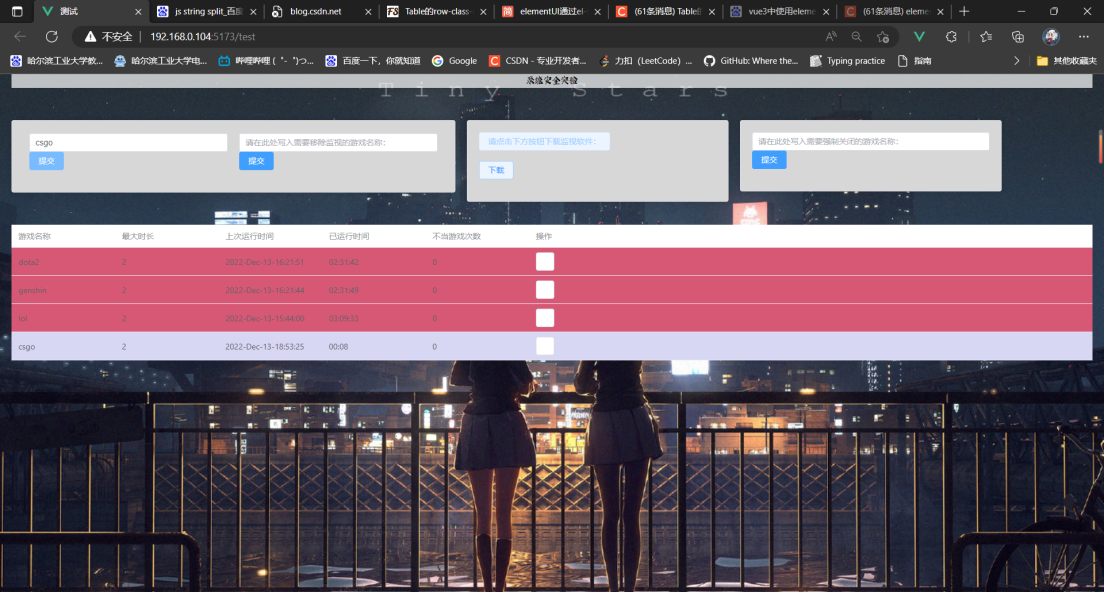


### 2.2.2 游戏列表管理：

左侧表单框中有两组输入框+按钮的组合，分别实现了需要监视游戏的添加与删除。

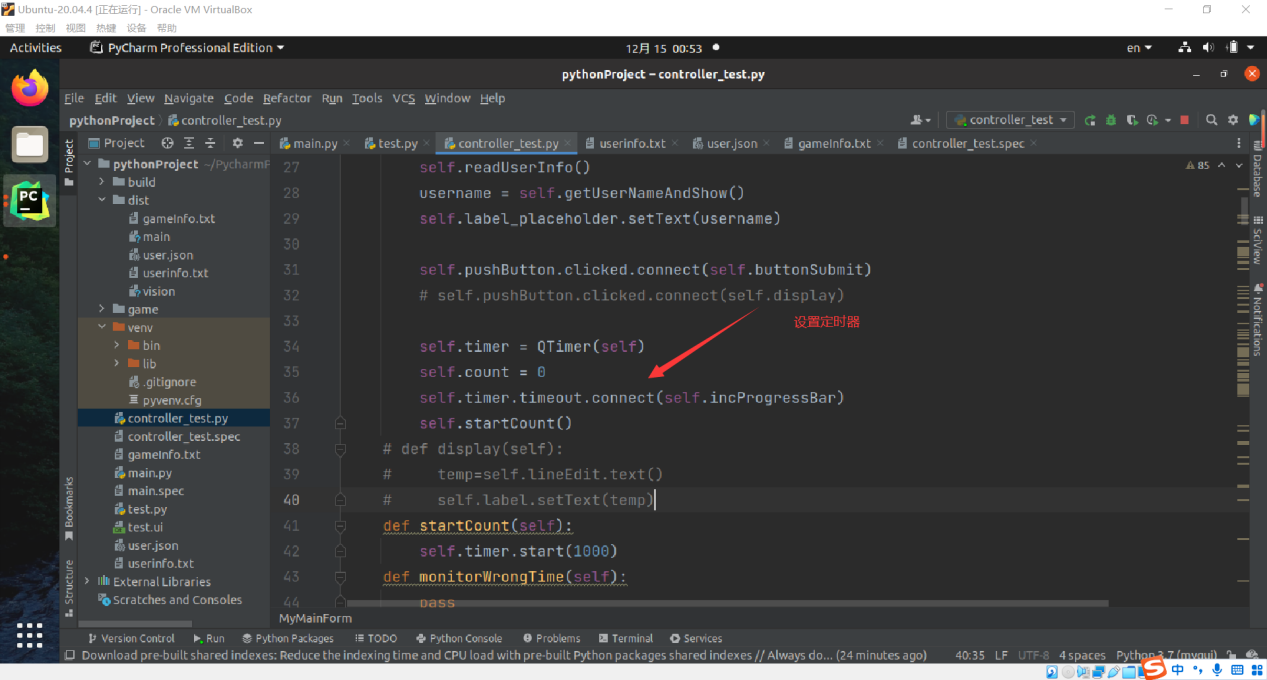


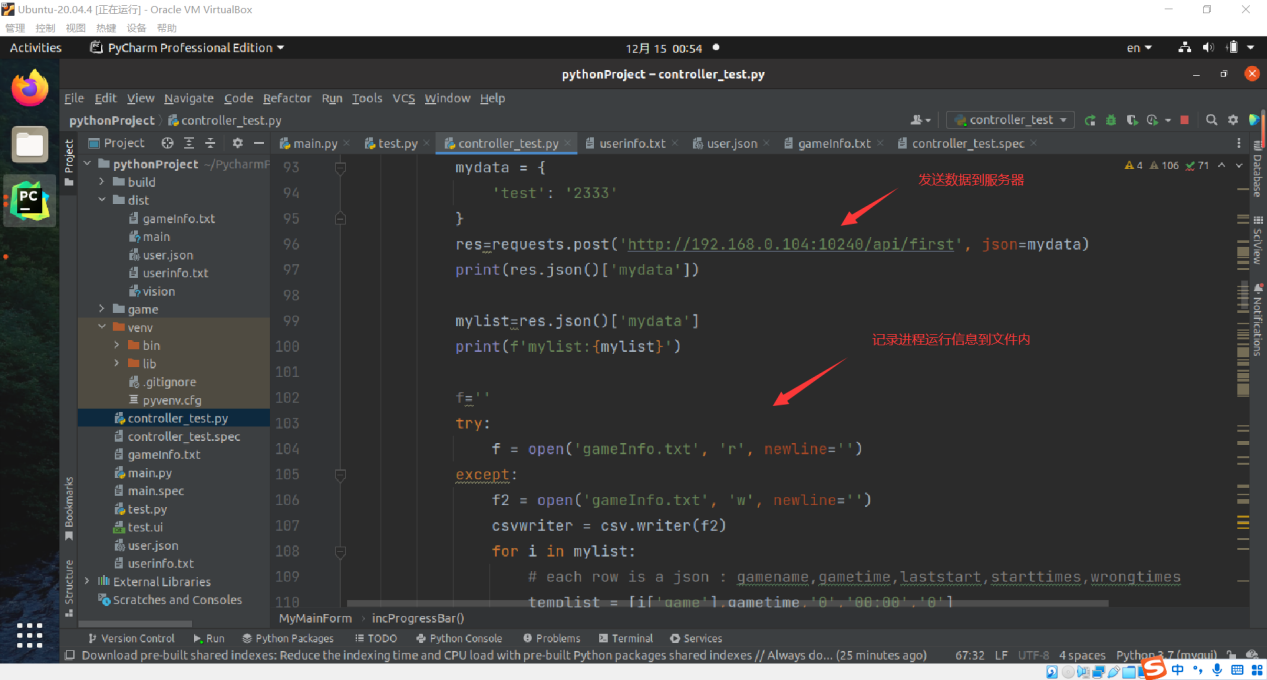
另外，在该页面中，我们记录了每款需要监视的游戏的最后登陆时间，同时根据用户在配置信息中所填的信息，规定了最多的游戏时长。比如此处，如果游戏时长多于2小时的话，该行条目就会变红，以示告警：



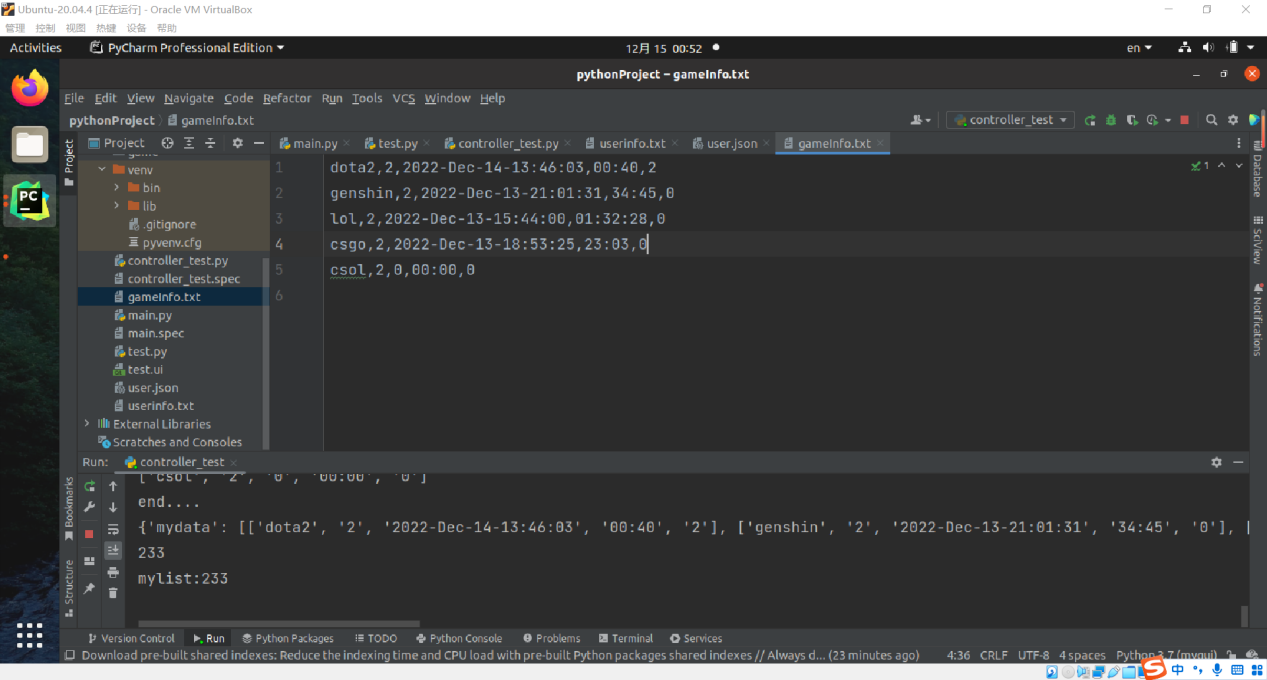
### 2.2.3 游戏运行检测：

在pyqt中，我们可以直接使用它的定时器模块，以做到定时检测游戏运行情况与发送信息。相关的代码见下图。注意到我们在python中使用requests库发送http请求到服务器，使用subprocess库进行ubuntu的命令行窗口调用。





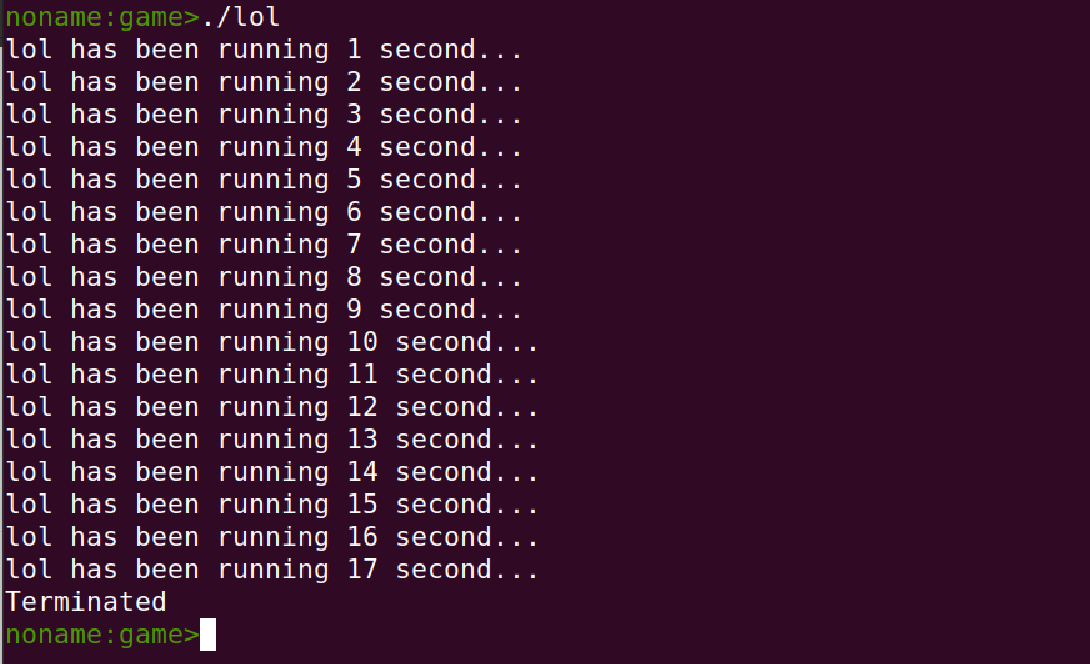
程序会每5秒，根据监控中心指定的游戏名称，在Ubuntu系统中进行查询，并且将查询到程序运行的总时间与最近一次运行的时刻记录在文件内，然后一并发送给服务器。服务器拿到数据之后发送到前端窗口，进行渲染到列表中。



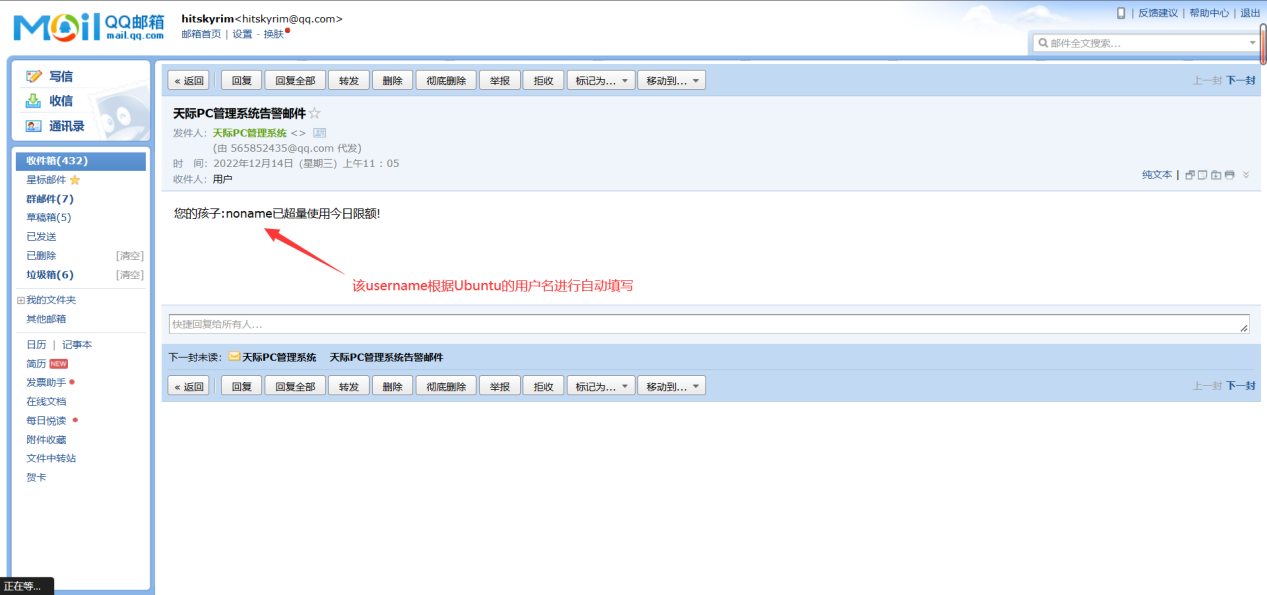
另外我们在前端页面中也提供了强制关闭某个进程的方法，即点击最后一列的按钮。



以lol为例，我们先在后台运行lol程序，接着在前台点击对应的白色按钮，即可强制关闭后台的lol程序，显示如下：

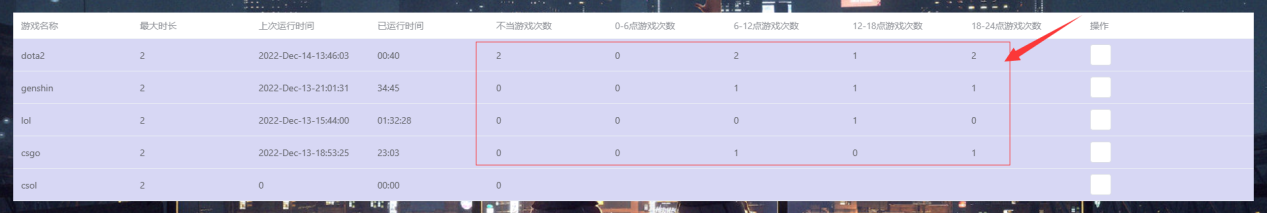


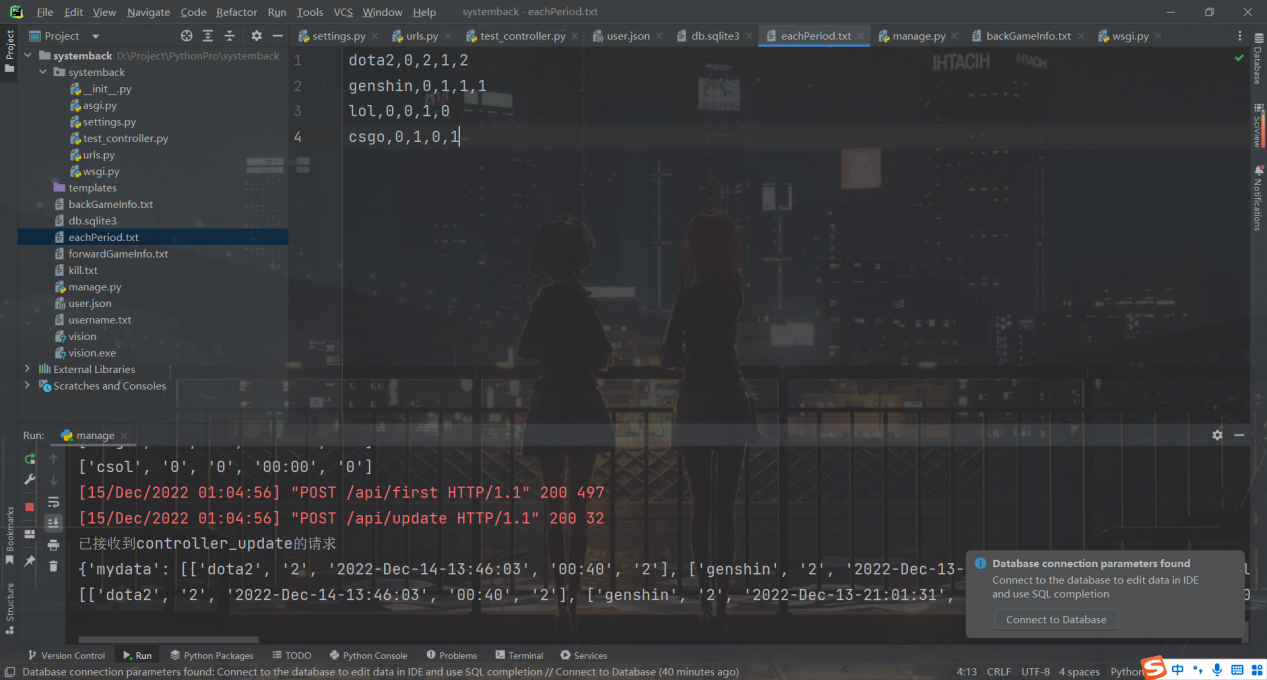
另外本程序还提供了告警的功能，即如果运行的时间超过所允许的最大时间的话，会给父母的邮箱发送邮件提醒，功能呈现如下：



### 2.2.4 游戏运行检测：

我们将每天分为四个时间段，并统计了各个时间段游戏的运行次数。相关的数据都填写在服务器的eachPeriod.txt文件中。





### 2.2.5 日志上报接口

在本系统中可以生成文件，以供管理部分进行分析评估。另外本系统也提供了一些后台接口，可以供更多页面查询使用。

### 2.2.6 其他问题

（1）是否可以绕开身份追踪检查，如工作时间以父母的账号登录等。

本次实验主要采用的是监测Ubuntu本机的用户名，所以如果青少年知晓自己父母的Ubuntu用户的账号与密码，即可实现登陆。

（2）对于绕开身份追踪检查的行为列出可能的检测方法。

如每次登陆需要在本机监测软件内输入密码，以确认身份的真实性。

# 分工

|  |  |
| --- | --- |
| 用户身份管理部分 | 吴昊 |
| 游戏列表管理部分 | 王泽鑫 |
| 游戏运行检测部分 | 刘昕烨 |
| 日志上报接口/前后端设计（vue+django） | 吴昊，袁野 |
| pyqt程序开发 | 吴昊 |
| 报告撰写 | 袁野，吴昊，王泽鑫，刘昕烨 |

# 总结

在这次实验中，我们小组不仅丰富了项目开发的经验，拥有了更深刻的对于Python各种库函数以及Ubuntu命令行指令的理解。比如本次实验中要求我们使用CS的设计模式，以监控各个用户的游戏运行情况，为了处理这种情况，我们想到了使用pyqt包进行图形化界面的设计，并且使用requests包定时向服务发送数据，以实现服务器一对多地处理数据，客户端多对一地上传数据的功能。

另外，我们使用vue快速搭建了一个前端页面，以方便用户进行处理与浏览。并且使用Django开发了一个小型的服务器，以方便我们处理前端页面和pyqt程序发来的信息，并且方便了我们对数据进行归档。

subprocess库在我们的实验中起到了重要的作用，这个库起到了沟通python与ubuntu命令行的作用，非常的方便快捷。

最后感谢大家的努力与付出，才能完成这一项实验。