



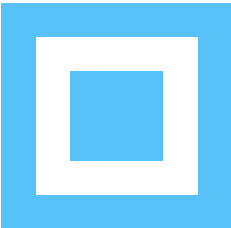
Public PC Guard



Team 4. Geonryun Lee
Yerim Chu

Jaekyung You
Woojoo You

Kyungtae Kang



Index

1. Overview

- Why we need our program
- Team member role

2. Purpose

- The effect we get from our program

3. Contents

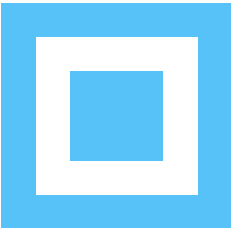
- Schematic diagram
- Application detail
- PC remote control detail
- Server and DB Management detail
- Log transport detail

4. Summery

- Future plan
- post-launch effect

Public PC Guard

1. Overview

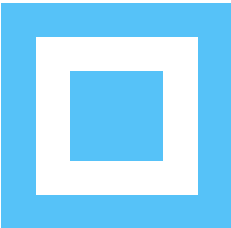


1. Overview

1) Why we need our program

Who?



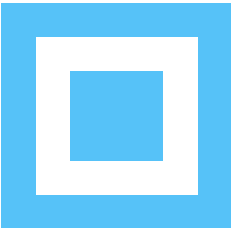


1. Overview

1) Why we need our program

You!





1. Overview

2) Team member role



Geonryun

Project Manager



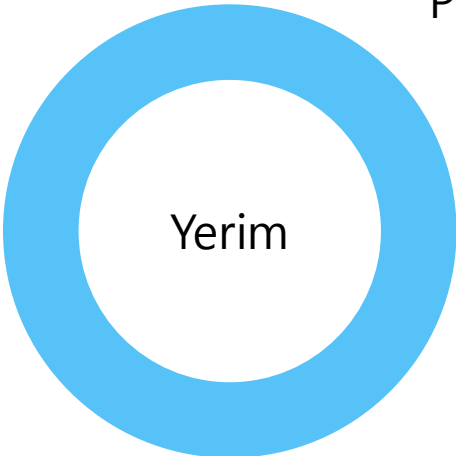
Jaekyung

PC remote control



Kyungtae

Manage server & DB



Yerim

Log transport &
Create presentation

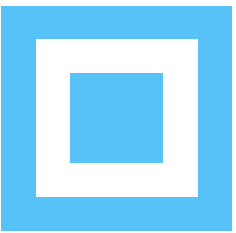


Woojoo

Development
android app

Public PC Guard

2. Purpose



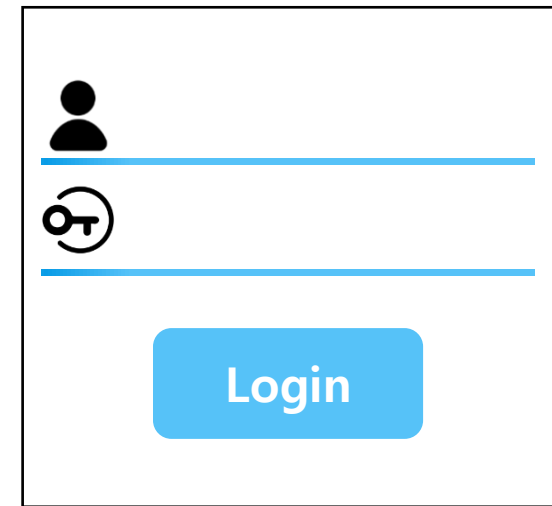
2. Purpose

- 1) The effect we get from our program



This prevents the use of the PC indiscriminately.

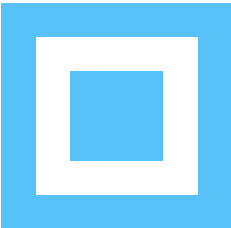
Responsibility



This reinforces responsibility through login authentication.

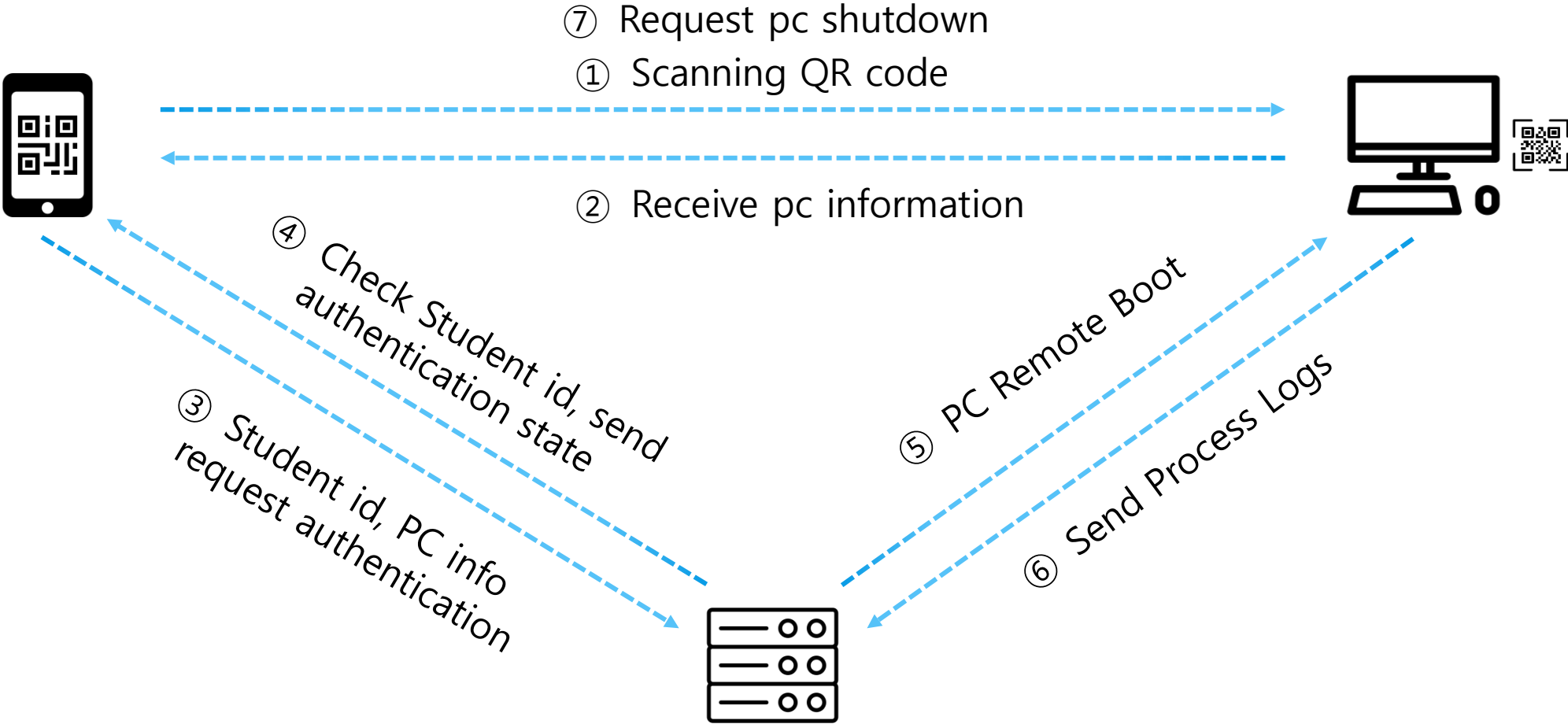
Public PC Guard

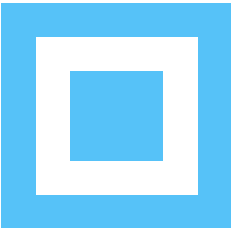
3. Contents



3. Contents

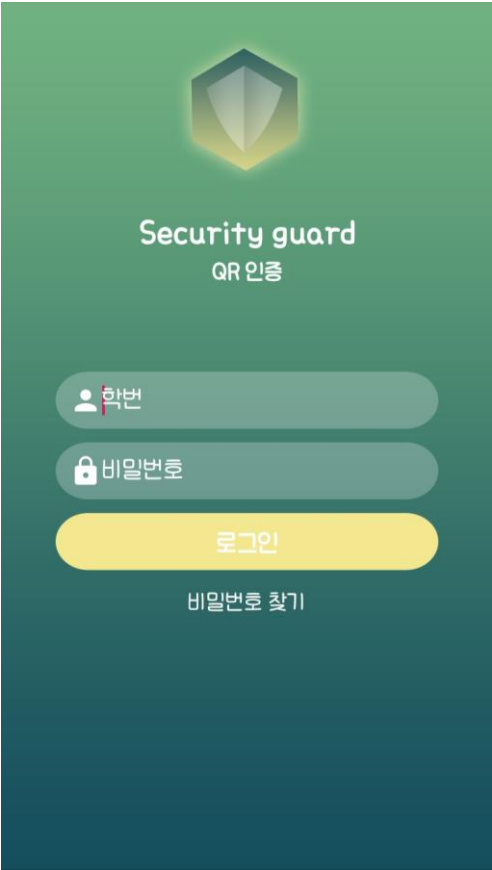
1) Schematic diagram



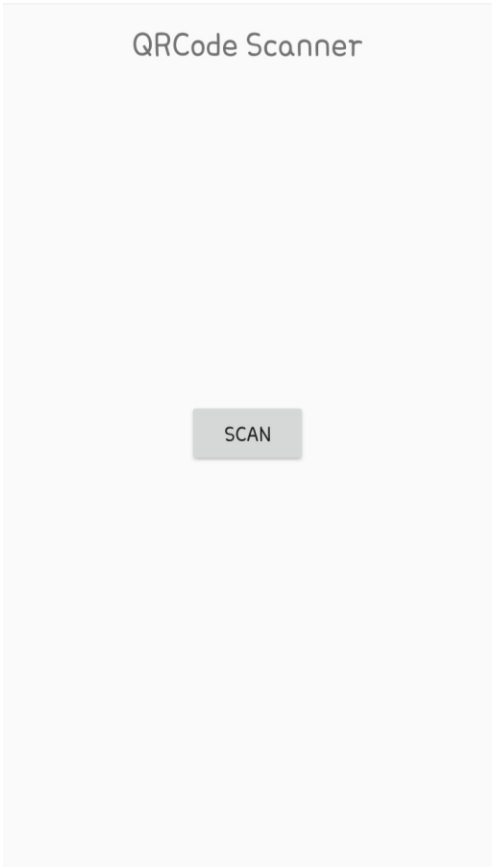


3. Contents

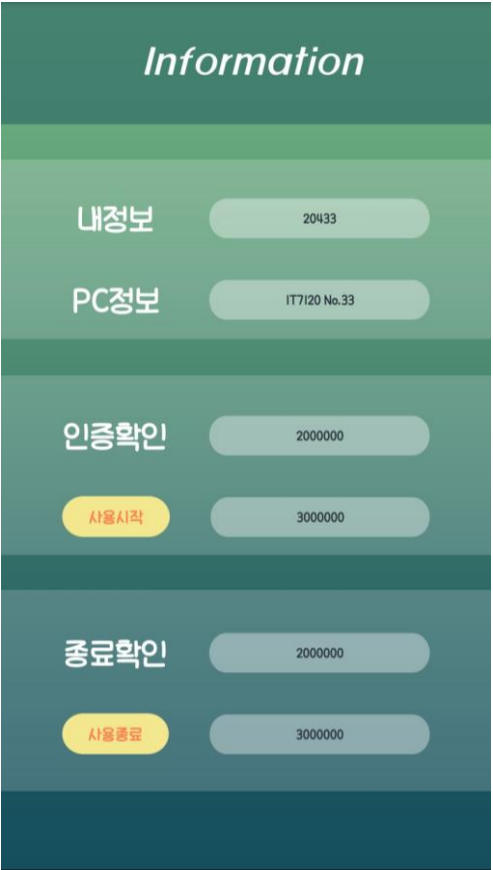
2) Application detail



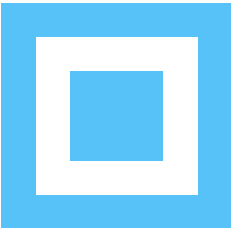
Log-in



Scanning QR code

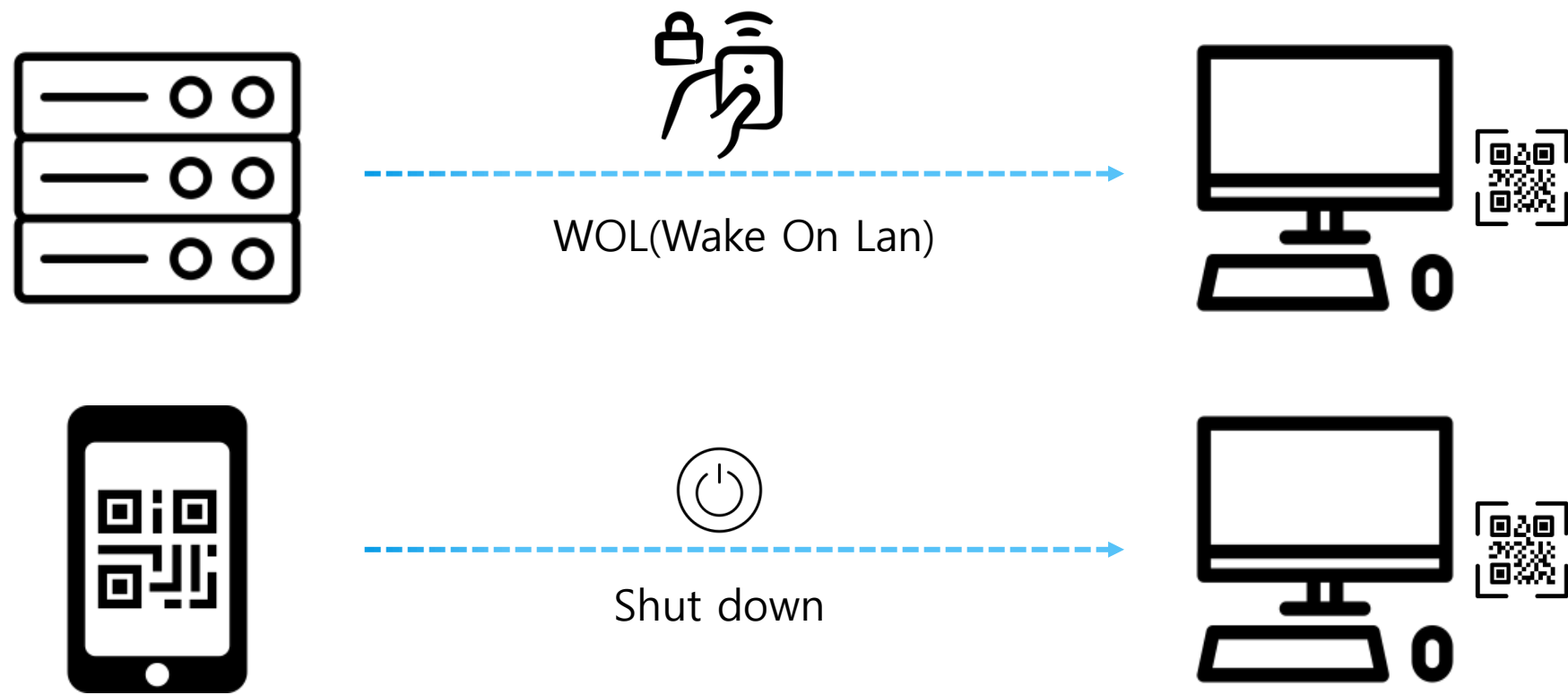


PC Information



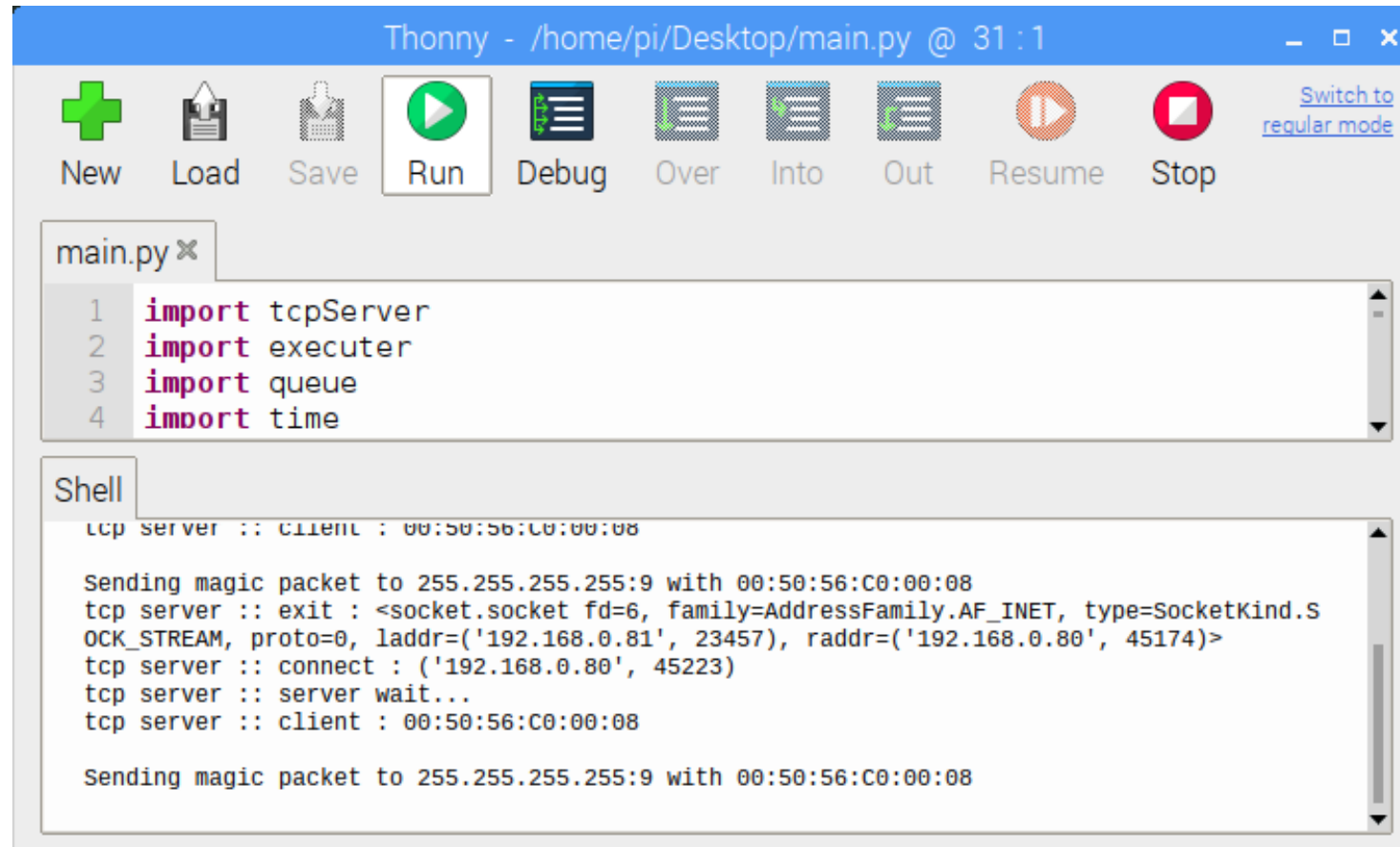
3. Contents

3) PC remote control detail



3. Contents

3) PC remote control detail



The screenshot shows the Thonny Python IDE interface. The title bar reads "Thonny - /home/pi/Desktop/main.py @ 31 : 1". The toolbar includes icons for New, Load, Save, Run (highlighted with a red box), Debug, Over, Into, Out, Resume, and Stop. A link "Switch to regular mode" is visible on the right. The editor window shows a Python script named "main.py" with the following code:

```
1 import tcpServer
2 import executer
3 import queue
4 import time
```

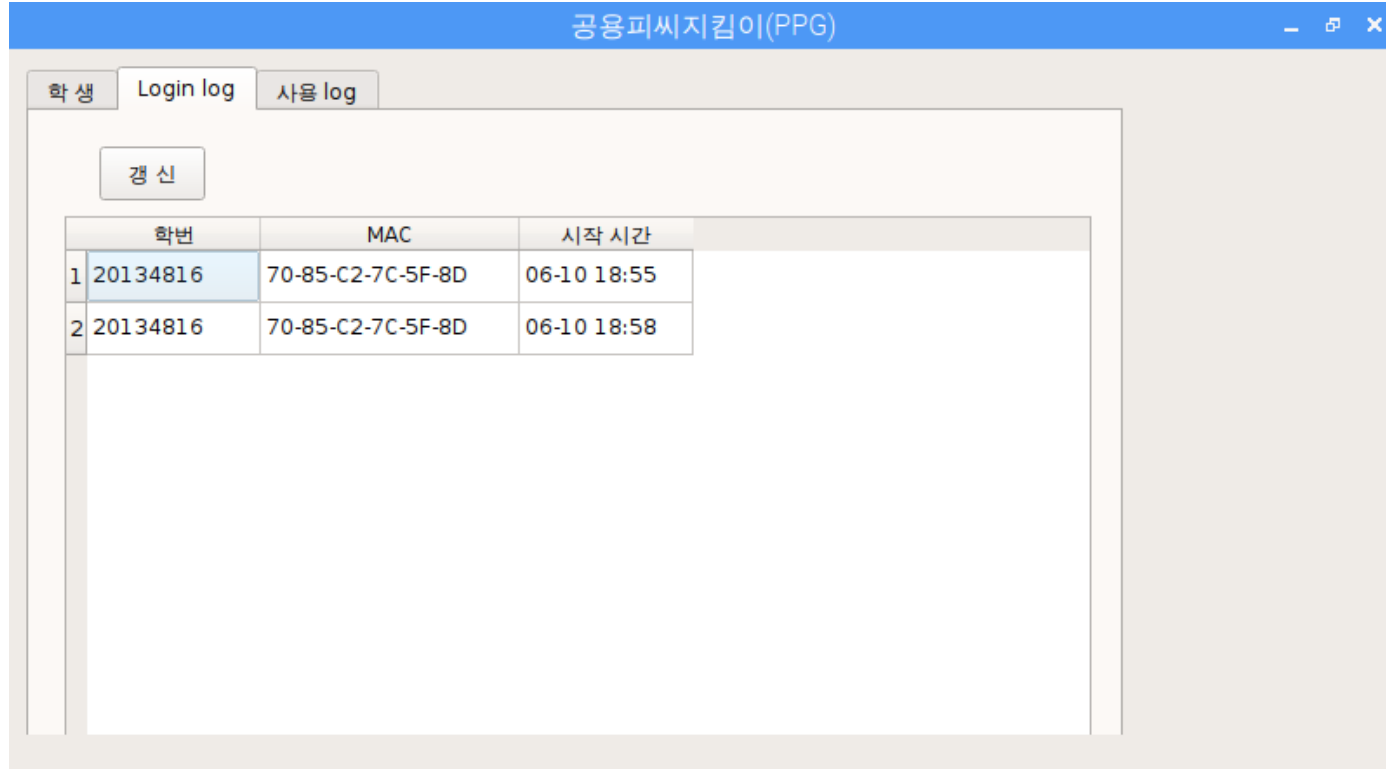
Below the editor is a "Shell" window showing the execution output:

```
tcp server :: client : 00:50:56:C0:00:08
Sending magic packet to 255.255.255.255:9 with 00:50:56:C0:00:08
tcp server :: exit : <socket.socket fd=6, family=AddressFamily.AF_INET, type=SocketKind.SOCK_STREAM, proto=0, laddr=('192.168.0.81', 23457), raddr=('192.168.0.80', 45174)>
tcp server :: connect : ('192.168.0.80', 45223)
tcp server :: server wait...
tcp server :: client : 00:50:56:C0:00:08
Sending magic packet to 255.255.255.255:9 with 00:50:56:C0:00:08
```

With communication between pc and server, remote boot was realized using wol function.

3. Contents

4) Server and DB Management detail



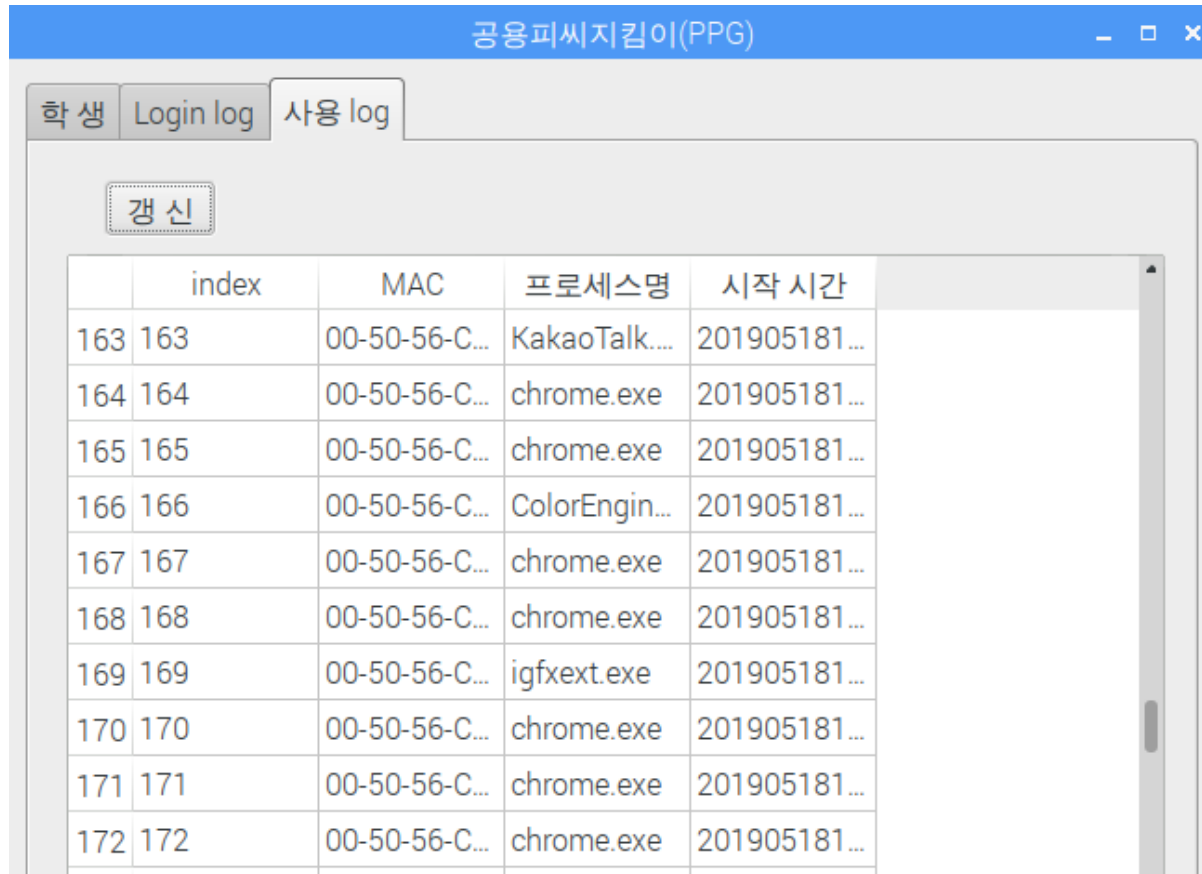
The screenshot shows a Python GUI application window titled '공용피씨지킴이(PPG)'. It has three tabs: '학생' (Student), 'Login log' (selected), and '사용 log' (Usage log). Below the tabs is a '검색' (Search) button. A table displays login logs with columns '학번' (Student ID), 'MAC', and '시작 시간' (Start Time). The table contains two rows of data.

	학번	MAC	시작 시간
1	20134816	70-85-C2-7C-5F-8D	06-10 18:55
2	20134816	70-85-C2-7C-5F-8D	06-10 18:58

We created gui with python to make database management convenient.

3. Contents

5) Log transport detail



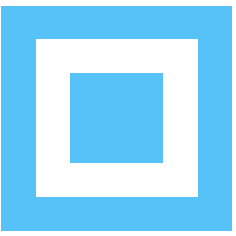
The screenshot shows a web application window titled "공용피씨지킴이(PPG)". It has three tabs: "학생" (Student), "Login log", and "사용 log" (Usage Log). The "사용 log" tab is active, showing a table of process execution logs. Above the table is a button labeled "갱신" (Refresh). The table has five columns: an index column, an "index" column, a "MAC" column, a "프로세스명" (Process Name) column, and a "시작 시간" (Start Time) column. The data rows show various processes like KakaoTalk, chrome.exe, ColorEngin, and igfxext.exe, all with a start time of 201905181...

	index	MAC	프로세스명	시작 시간
163	163	00-50-56-C...	KakaoTalk...	201905181...
164	164	00-50-56-C...	chrome.exe	201905181...
165	165	00-50-56-C...	chrome.exe	201905181...
166	166	00-50-56-C...	ColorEngin...	201905181...
167	167	00-50-56-C...	chrome.exe	201905181...
168	168	00-50-56-C...	chrome.exe	201905181...
169	169	00-50-56-C...	igfxext.exe	201905181...
170	170	00-50-56-C...	chrome.exe	201905181...
171	171	00-50-56-C...	chrome.exe	201905181...
172	172	00-50-56-C...	chrome.exe	201905181...

Processes executed by the user were collected and recorded in the database.

Public PC Guard

4. Summery



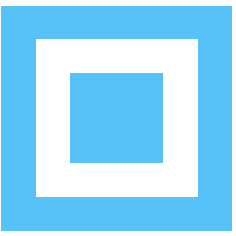
4. Summery

1) Future plan



In collecting logs, we will use machine learning and deep learning to precisely collect logs.

It will also add a function to learn the malicious process and alert the administrator if it is on database record.



4. Summery

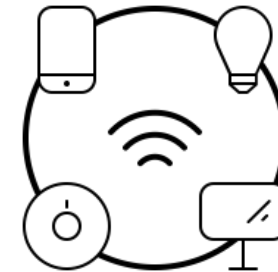
2) post-launch effect



It can be useful in schools or public institutions that have public pc rooms.



The cost of administration will be reduced by reducing the number of people that the agency manages public rooms.



The program is suitable for IoT and can be applied to smart home systems, not just to manage public pc rooms.



Thank you for Listening!

Contact : https://github.com/chuyr/Cap4team_2019

