



Course Timetable Inquiry

Application mobile vision:

Mobile Computing



Here are 彭容微2030026116, 钱景彬2030026117, 冯子骜2030026036



01 Introduction & Background

Environment Built / Database Designed

02 Implementation

Server / Client / Functions / Concurrency connections

03 Demo Presentation

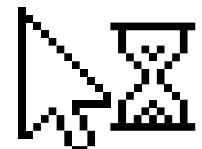
Show the whole program



01 Introduction/ Environment built

Analyze the Assignment, build
the environment and database.

彭容微2030026116





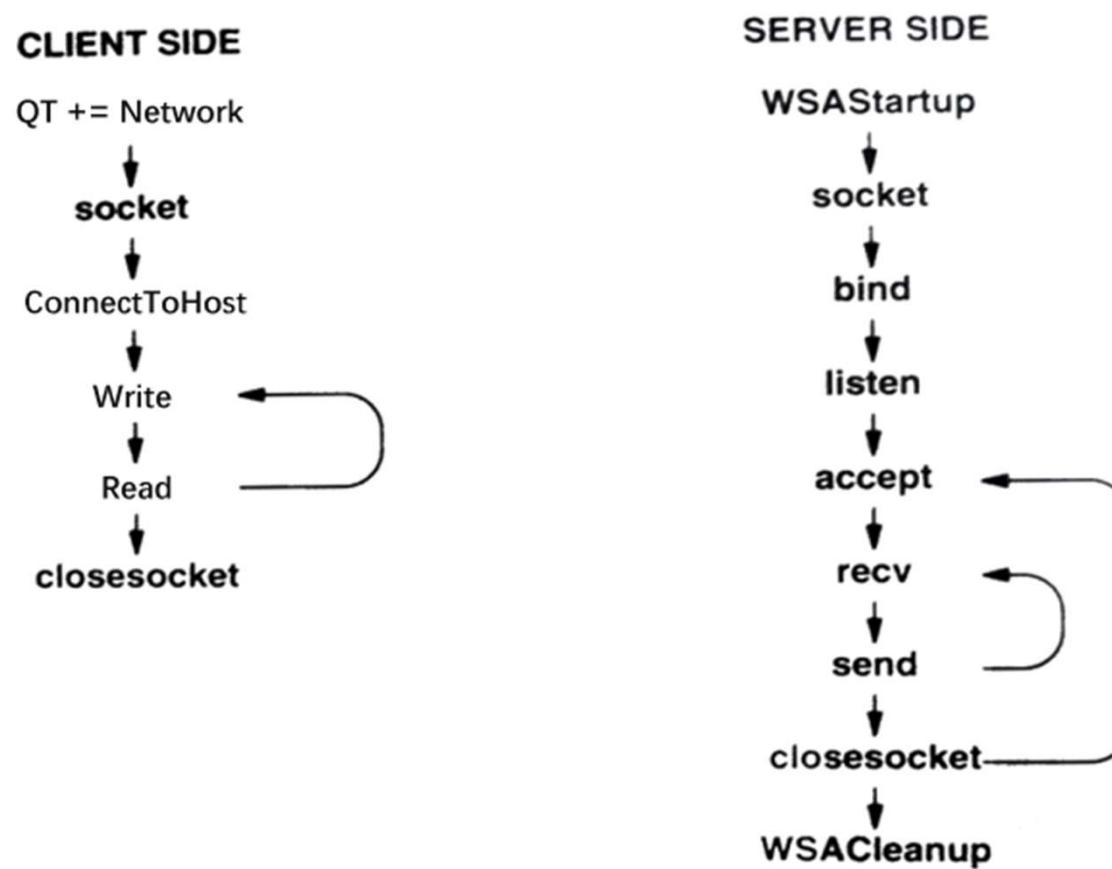
II. Requirements

Students are required to design and implement an application of Course Timetable Inquiry in mobile platforms (e.g., Android OS) using the Qt development tool. Two sample programs (namely, “server.cpp” and “client.cpp”) are given to students for reference (students can download them from iSpace). The program should consist of the following modules:

- (1) Database or text file: to store the course information (e.g., course code, course title, section, time, classroom, instructor);
- (2) User management: to allow the administrator to add, update, or delete users;
- (3) Information update: to allow the administrator to update parts of a course’s timetable;
- (4) Connections: to support at least three concurrent connections;
- (5) System management: to provide historical connection statistics to the administrator, such as IP address, connection time, etc.



Frame for client-server communication



A table for course information

+ 选项

← →

MID

CourseCode

CourseTitle

Session

Classroom

Time

Instructor

<input type="checkbox"/>		编辑		复制		删除	1	ACCT2003	Principles of Accounting I	1001	T5-308	Wed 17:00-17:50	Dr. Mei Chun KONG
<input type="checkbox"/>		编辑		复制		删除	2	ACCT2003	Principles of Accounting I	1001	T7-405	Fri 18:00-18:50	Dr. Mei Chun KONG
<input type="checkbox"/>		编辑		复制		删除	3	ACCT2003	Principles of Accounting I	1006	T7-602	Wed 12:00-12:50	Prof. Lei SHI
<input type="checkbox"/>		编辑		复制		删除	4	ACCT2003	Principles of Accounting I	1006	T2-609	Thu 8:00-8:50	Prof. Lei SHI
<input type="checkbox"/>		编辑		复制		删除	5	ACCT2003	Principles of Accounting I	1002	T8-302	Thu 14:00-15:50	Dr. Ge SONG
<input type="checkbox"/>		编辑		复制		删除	6	ACCT2003	Principles of Accounting I	1002	T2-207	Tue 14:00-14:50	Dr. Ge SONG
<input type="checkbox"/>		编辑		复制		删除	7	ACCT2003	Principles of Accounting I	1004	T2-604	Wed 10:00-11:50	Dr. Tenglong WAN
<input type="checkbox"/>		编辑		复制		删除	8	ACCT2003	Principles of Accounting I	1004	T2-407	Mon 15:00-15:50	Dr. Koun CHOI
<input type="checkbox"/>		编辑		复制		删除	9	ACCT2003	Principles of Accounting I	1005	T2-101	Tue 15:00-16:50	Dr. Koun CHOI
<input type="checkbox"/>		编辑		复制		删除	10	ACCT2003	Principles of Accounting I	1005	T2-512	Wed 11:00-11:50	Dr. Sherry ZHOU
<input type="checkbox"/>		编辑		复制		删除	11	ACCT2003	Principles of Accounting I	1007	T2-104	Thu 15:00-16:50	Dr. Sherry ZHOU



A table for storing users

+ 选项

← →

				user_name	priority
<input type="checkbox"/>				admin1	2
<input type="checkbox"/>				admin2	2
<input type="checkbox"/>				admin3	2
<input type="checkbox"/>				admin4	2
<input type="checkbox"/>				admin5	2
<input type="checkbox"/>				admin6	2
<input type="checkbox"/>				admin7	2
<input type="checkbox"/>				adqe	1
<input type="checkbox"/>				ansaier	1
<input type="checkbox"/>				baba2a	1
<input type="checkbox"/>				bababa	1
<input type="checkbox"/>				bugtest	1
<input checked="" type="checkbox"/>				buqtest2	1

We can add other priority for further use.

If priority = 1, then this user is a student.

If priority = 2, then this user is a administrator.



A table for collecting the historical statistic

+ 选项

	<input type="button" value="← T →"/>	MID	1	USER_NAME	IP_ADDR	CONNECTION_TIME	RUN_TIME
<input type="checkbox"/>	编辑	复制	删除	171	admin1	192.168.136.118	2022-12-4 21:39:54
<input type="checkbox"/>	编辑	复制	删除	170	admin1	192.168.136.118	2022-12-4 21:36:20
<input type="checkbox"/>	编辑	复制	删除	169	admin1	192.168.136.118	2022-12-4 21:34:34
<input type="checkbox"/>	编辑	复制	删除	168	admin2	192.168.136.118	2022-12-4 21:24:31
<input type="checkbox"/>	编辑	复制	删除	167	admin1	192.168.136.240	2022-12-4 16:43:50
<input type="checkbox"/>	编辑	复制	删除	166	admin1	192.168.136.240	2022-12-4 15:49:53
<input type="checkbox"/>	编辑	复制	删除	165	admin1	192.168.136.240	2022-12-4 15:37:44
<input type="checkbox"/>	编辑	复制	删除	164	user1	192.168.136.240	2022-12-4 15:37:9
<input type="checkbox"/>	编辑	复制	删除	163	user1	192.168.136.240	2022-12-4 15:32:57
<input type="checkbox"/>	编辑	复制	删除	162	user1	192.168.136.240	2022-12-4 15:21:26
<input type="checkbox"/>	编辑	复制	删除	161	admin1	127.0.0.1	2022-11-22 22:28:4
<input type="checkbox"/>	编辑	复制	删除	160	admin1	127.0.0.1	2022-11-16 14:47:16
<input checked="" type="checkbox"/>	控制台	编辑	复制	159	user1	127.0.0.1	2022-5-23 9:23:3

Using XAMPP and MariaDB Connector to connect database

The screenshot shows the XAMPP Control Panel v3.2.4 interface. On the left, there's a sidebar with icons for modules like Apache, MySQL, FileZilla, Mercury, and Tomcat. The main area displays the status of Apache (PID 38076) and MySQL (PID 21748). Apache is running on port 4433, 8081, with actions for Stop, Admin, Config, and Logs. MySQL is running on port 3306, with actions for Stop, Admin, Config, and Logs. To the right of the main panel is a file explorer window titled "Data (D:) > Qt > 5.15.2 > Src > qtbase > src > plugins > sqldrivers > mysql >". It lists files and folders related to the MariaDB Connector, including "include", "lib", "libmysql", "main.cpp", "mysql.json", "mysql.pro", "mysql.pro.user", "qsql_mysql.cpp", "qsql_mysql_p.h", and "README". The "main.cpp" file is identified as a C++ Source file.

名称	修改日期	类型	大小
include	2022/11/22 20:38	文件夹	
lib	2022/11/22 20:38	文件夹	
libmysql	2022/11/22 20:58	文件夹	
main.cpp	2020/10/27 16:02	C++ Source file	3 KB
mysql.json	2020/10/27 16:02	JSON 文件	1 KB
mysql.pro	2022/11/22 20:40	Qt Project file	1 KB
mysql.pro.user	2022/11/23 18:42	USER 文件	45 KB
qsql_mysql.cpp	2020/10/27 16:02	C++ Source file	50 KB
qsql_mysql_p.h	2020/10/27 16:02	H 文件	4 KB
README	2020/10/27 16:02	文件	1 KB

And the MariaDB location to mysql.pro Recompile the mysql project in Qt

The screenshot shows the Qt Creator interface with the following details:

- Title Bar:** mysql.pro @ mysql - Qt Creator
- Menu Bar:** 文件(F) 编辑(E) View 构建(B) 调试(D) Analyze 工具(I) 控件(W) 帮助(H)
- Toolbars:** 欢迎, 编辑, 设计, Debug, 项目, 帮助
- Project Explorer:** 显示了项目的文件结构:
 - mysql (包含 mysql.pro, qsqldriverbase.pri, Headers/qsql_mysql_p.h, Sources/main.cpp, Sources/qsql_mysql.cpp, Other files/mysql.json)
- Editor:** 显示了 mysql.pro 的内容，其中包含 MariaDB 相关的配置。

```
1 TARGET      = qsqlmysql
2
3 HEADERS     += $$PWD/qsql_mysql_p.h
4 SOURCES    += $$PWD/qsql_mysql.cpp $$PWD/main.cpp
5
6 # QMAKE_USE   += mysql
7
8 OTHER_FILES += mysql.json
9
10 PLUGIN_CLASS_NAME = QMYSQLDriverPlugin
11 include(../qsqldriverbase.pri)
12
13 LIBS        += -L'C:/Program Files (x86)/MariaDB/MariaDB Connector C/lib/' -llibmariadb
14 INCLUDEPATH += 'C:/Program Files (x86)/MariaDB/MariaDB Connector C/include'
15 DEPENDPATH  += 'C:/Program Files (x86)/MariaDB/MariaDB Connector C/lib'
16 DESTDIR     = ../mysql/lib/
17
```
- Output Panel:** 显示了多次错误消息，表明无法读取 qtsqldrivers-config.pri 文件。

```
Cannot read D:/qtsqldrivers-config.pri: No such file or directory
Cannot read D:/configure.pri: No such file or directory
Cannot read D:/qtsqldrivers-config.pri: No such file or directory
Cannot read D:/qtsqldrivers-config.pri: No such file or directory
Cannot read D:/qtsqldrivers-config.pri: No such file or directory
```
- Bottom Bar:** Type to locate (Ctrl+...), 1 问题, 2 Search Results, 3 应用程序输出, 4 编译输出, 5 QML Debugger..., 6 概要信息, 7 Test Results



Include this library to server

```
LIBS += -L'D:/Qt/5.15.2/Src/qtbase/src/plugins/sqldrivers/mysql/lib' -llibmariadb  
LIBS += -lwsock32  
INCLUDEPATH += 'D:/Qt/5.15.2/Src/qtbase/src/plugins/sqldrivers/mysql/include'  
DEPENDPATH += 'D:/Qt/5.15.2/Src/qtbase/src/plugins/sqldrivers/mysql/include'
```



Some settings in Qt Creator

设备

Devices Android QNX SSH

Android Settings

JDK location: D:\Java\jdk1.8.0_131\ 浏览...

Android SDK的路径: \Users\Steve\AppData\Local\Android\Sdk 浏览... Set Up SDK

SDK Manager

Android NDK list:

- 锁 C:\Users\Steve\AppData\Local\Android\Sdk\ndk\21.3.
- 锁 C:\Users\Steve\AppData\Local\Android\Sdk\ndk\23.1.

Add... 删除 Make Default

◀ ▶

✓ Android settings are OK. (SDK Version: 8.0, NDK Version: 21.3.6528147) 详情 ▾

Automatically create kits for Android tool chains

For building Android application



ReadMe.txt - 记事本

文件 编辑 查看

Before executing the program, you need to do following things:

1. Set up Qt environment

1.1 Set up MySQL database environment

Here, we use 32-bit MariaDB Connector C to connect the database.

1.1.1 Compile Qt MySQL driving

reset mysql.pro file in the install path of your own Qt.

(a) comment "QMAKE_USE += mysql"

(b) Open the program folder -> MariaDB Connector C ->

 Copy the path of include folder

 Copy the path of lib folder

(c) Paste lib folder path after "LIBS += -L'<libpath>' -llibmariadb"

 Example: win32: LIBS += -LC:/Program Files (x86)/MariaDB/MariaDB Connector C/lib/ -llibmariadb

(d) Paste include folder path after "INCLUDEPATH += '<include path>'"

 Example: INCLUDEPATH += 'C:/Program Files (x86)/MariaDB/MariaDB Connector C/include'

(e) Paste lib folder path after "DEPENDPATH += '<lib folder path>'"

 Example: DEPENDPATH += 'C:/Program Files (x86)/MariaDB/MariaDB Connector C/lib'

(f) Add the destination directory

 After last step, add one line of code:

 DESTDIR = ../mysql/lib/

(g) Use your desktop to compile and run this .pro file

 If success, a new folder called 'lib' will appear under mysql.pro current directory and go to 1.12.

1.1.2 Add mysql dynamic link libraries

(a) In last step lib path, copy two dll files under to mysql path

Path Example :D:/Qt/Qt5.12.10/5.12.10/Src/qtbase/src/plugins/sqldrivers/mysql/lib/

.dll files name: qsqlmysql.dll and qsqlmysqld.dll

1.1.3 Add mysql include files and libraries to Qt environment

Copy the files and folders in MariaDB Connector C and paste them under your Qt source mysql folder:

'/Qt/Qt5.12.10/5.12.10/Src/qtbase/src/plugins/sqldrivers/mysql/'

1.2 Set up server environment

行 1, 列 1

100%

Windows (CRLF)

UTF-8

ReadMe.txt - 记事本

文件 编辑 查看

1.2 Set up server environment

1.2.1 Add mysql library dependencies and winsocket libraries

Change all path to your own Qt:

LIBS += -LD:/Qt/Qt5.12.10/5.12.10/Src/qtbase/src/plugins/sqldrivers/mysql/lib' -llibmariadb

LIBS += -lwsock32

INCLUDEPATH += 'D:/Qt/Qt5.12.10/5.12.10/Src/qtbase/src/plugins/sqldrivers/mysql/include'

DEPENDPATH += 'D:/Qt/Qt5.12.10/5.12.10/Src/qtbase/src/plugins/sqldrivers/mysql/include'

1.2.2 Compile this server.pro in 32-bit Kits:

Here we recommend to use:

Desktop Qt 5.12.10 MinGW 32-bit

1.3 Set up client environment

1.3.1 open Qt Creator -> Tool -> Performance -> Device -> Android ->

set up Java Settings and Android Settings (including Android SDK and Android NDK)

click 'Apply'

2. Run projects

2.1 Open server.pro in Qt creator

Right click server.pro and click compile

Right click again and run the server

Remind: Compile this server.pro in 32-bit Kits, here we recommend to use: Desktop Qt 5.12.10 MinGW 32-bit

2.2 Open client in Qt creator

Open Project1_1128.pro in Qt Creator, this is client.

Right click Project1_1128.pro and click compile.

Deploy to Android Device:

Here, use kit that fit your test mobile devices.

We tested our project on Android Qt 5.12.10 Clang arm64-v8a

Right click again and run the client.

2.3 Open USB debug Mode on your mobile phone when deploy this project.

Remember to use USB line to connect your mobile phone to your Desktop.

行 1, 列 1

100%

Windows (CRLF)

UTF-8

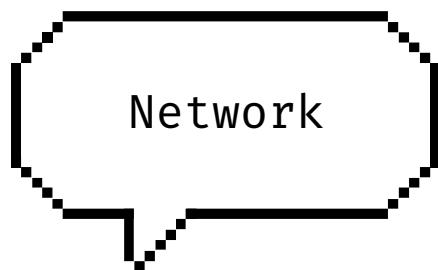


02

Implementation

How to implement the communications?

钱景彬2030026117



Network



Rough Outline

Client.cpp

4

Send the results back to Client. The Client will print these results.

1

Send the requests to Server.

2

Convert the requests and send SQL to database.

Server.cpp

3

Fetch the result from MYSQL database.





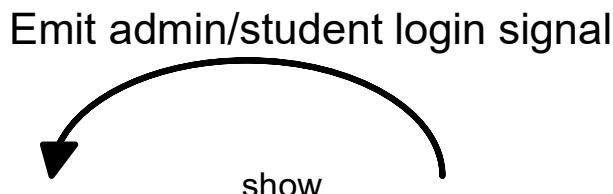
Detail Outline

Client

```
Main.cpp show
QString acti = "1";
acti.append(";").append(stuname.toStdString().data());
qDebug("acti: %s",acti.toStdString().data());
//search database students

//send socket to server
mSocket->flush();
mSocket->write(acti.toStdString().data(),acti.toStdString().length())
mSocket->flush();
```

information to database





Detail Outline

Server

Wait and listen for clients to connect

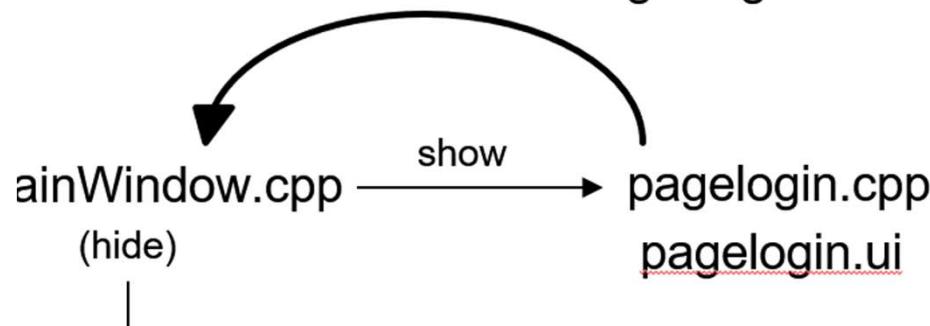


Receive the user's name from client
and search/send back the priority
from database.



priority=2 priority=1

Emit admin/student login signal



Send username to server and server gets the priority from database.



Server

Wait and listen for clients to connect



Receive the user's name from client
and search/send back the priority
from database.



Receive ActionType and
search/modified information from
client.
Do the split and use these
information to search on database.
Send back the result to client.

```
while(1)
{
    memset(szBuff, '\0', sizeof(szBuff));

    msg_len = recv(currThread->sock, szBuff, sizeof(szBuff)

    //if actionType is 0, szBuff has two strings, so we s

    if (szBuff[0]-48 == 0)
        convertReceive(szBuff, 2);
    //if actionType is 1, szbuff have two strings.
    if (szBuff[0]-48 == 1){
        convertReceive(szBuff, 2);
        // need to change to int, later sql add 1
    }
    //action type is 2
    if (szBuff[0]-48 == 2)
        convertReceive(szBuff, 3);
    //action tipe is 3
    if (szBuff[0]-48 == 3)
        convertReceive(szBuff, 2);
    //action type is 4
    if (szBuff[0]-48 == 4)
        convertReceive(szBuff, 4);
    if (szBuff[0]-48 == 5)
        convertReceive(szBuff, 2);
    //action type is 9
    if (szBuff[0]-48 == 9){
        printf("test1\n");
        convertReceive(szBuff, 4);
        printf("test2\n");
    }
}
```

7 action types for searching
courses, searching student,
adding student, deleting
students, updating student,
updating courses, and exit.



Server

```
switch(actionType){  
//case 0 is for student searching the course  
case 0:  
    //make sql string  
    char SQLfindCourses[100];  
    strcpy(SQLfindCourses, "select * from course where CourseTitle like '%");  
    strcat(SQLfindCourses, receiveMessage[0]);  
    strcat(SQLfindCourses, "%'");  
    //send course title to database  
    if(mysql_query(con, SQLfindCourses))  
    {  
        fprintf(stderr, "Client %d: Failed %s\n", currThread->t_id, mysql_error(con));  
        exit(EXIT_FAILURE);  
    }  
    res=mysql_store_result(con);  
    //num_fields= mysql_num_fields(res);  
    field=mysql_fetch_fields(res);  
    row = mysql_fetch_row(res);  
    int count;  
    count = res->row_count;  
    . . .
```

Use switch to enter different action types.

Add the message received from client to SQL quires.

Find the result from database.



Server

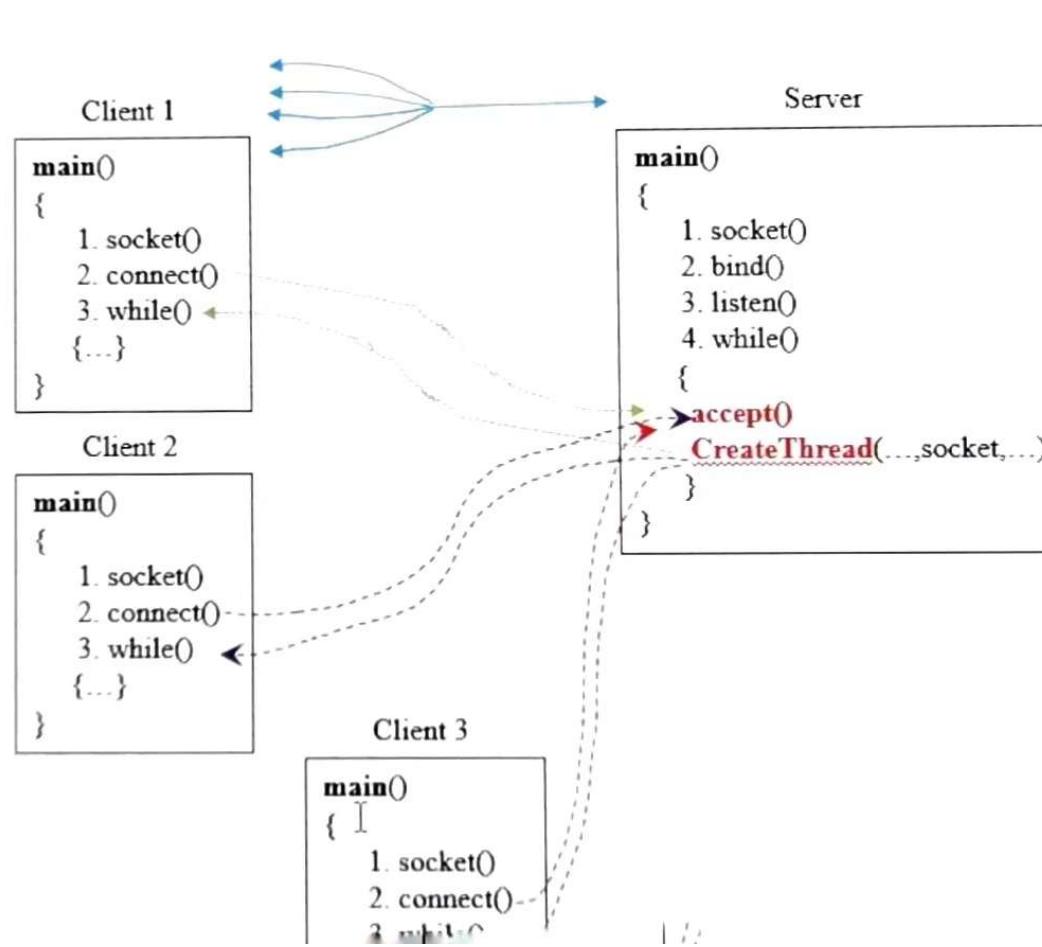
Send the results back to client.

```
while (row = mysql_fetch_row(res)){💡    △Using the result of an assignment as a condition without  
    sprintf(sendCourse, "%s, %s, %s, %s, %s, %s, %s", row[0], row[1], row[2], row[3], row[4], row[5]);  
    sleep(100);  
    printf("%s\n", sendCourse);  
    msg_len = send(currThread->sock, sendCourse, sizeof(sendCourse), 0);  
    memset(sendCourse, '\0', sizeof(sendCourse));  
}
```



Server

As for concurrent connections





Server

As for concurrent connections

```
//create threads for 3 connections
for(int i=0; i<4; i++){

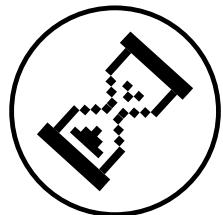
    addr_len = sizeof(client_addr);
    msg_sock = accept(sock, (struct sockaddr*)&client_addr, &addr_len);
    if (msg_sock == INVALID_SOCKET){
        fprintf(stderr, "accept() failed with error %d\n", WSAGetLastError());
        WSACleanup();
        return -1;
    }
    char client_address[20];
    _snprintf(client_address, 20, "%s", inet_ntoa(client_addr.sin_addr));
    printf("Client %d accepted connection from %s, port %d\n", i+1, client_address

    threadNeed[i].sock = msg_sock;
    threadNeed[i].t_id = i+1;
    threadNeed[i].count = 0;

    strcpy(threadNeed[i].ip, client_address);
    //create thread for each connection and go to database to fetch data.
    _beginthread(fetchDB, 0, &threadNeed[i]);
}
_endthread();
```

```
struct thread{
    SOCKET sock;
    int t_id;//thread id
    char ip[20];//ip address for client
    int count;
};

typedef thread Value;
```



03

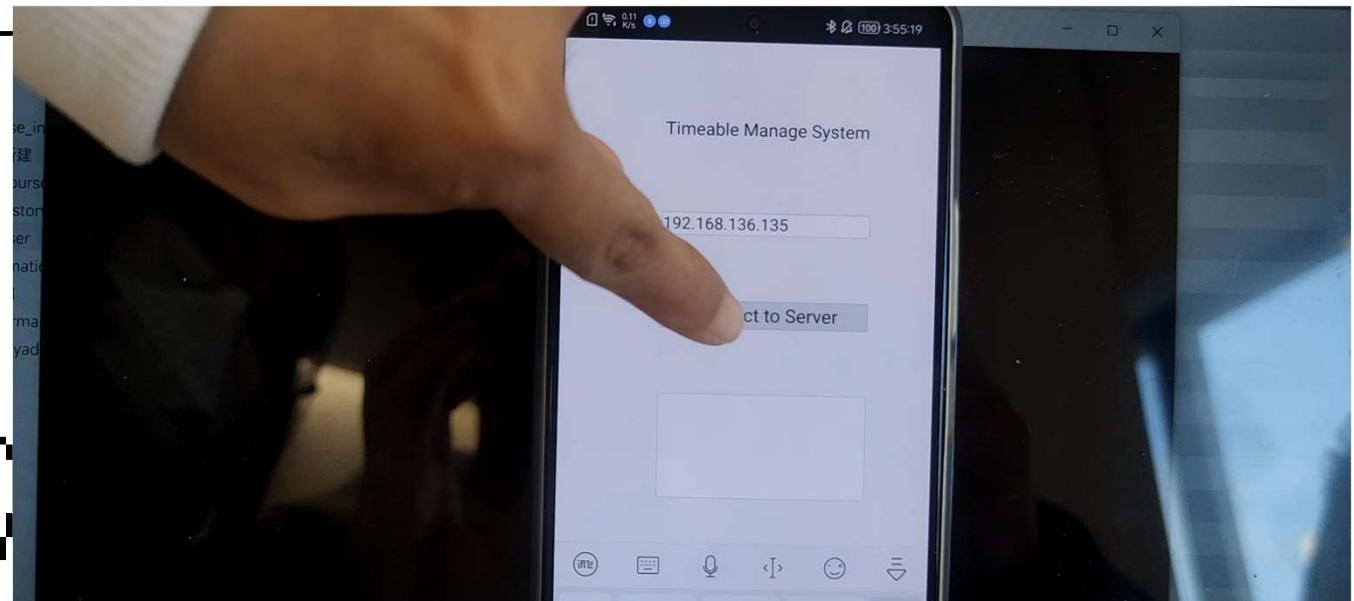
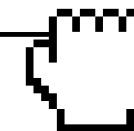
Demonstration

冯子骜2030026036



Connect to Server

User inputs the IP address, and the client will connect to the server.



0.92
K/s

100 2:10:33

User login

Timeable Manage System

192.168.136.135

Connect to Server

2022-12-4 14:10:27
Connect Success!

User Name:

Password:

Login

Exit

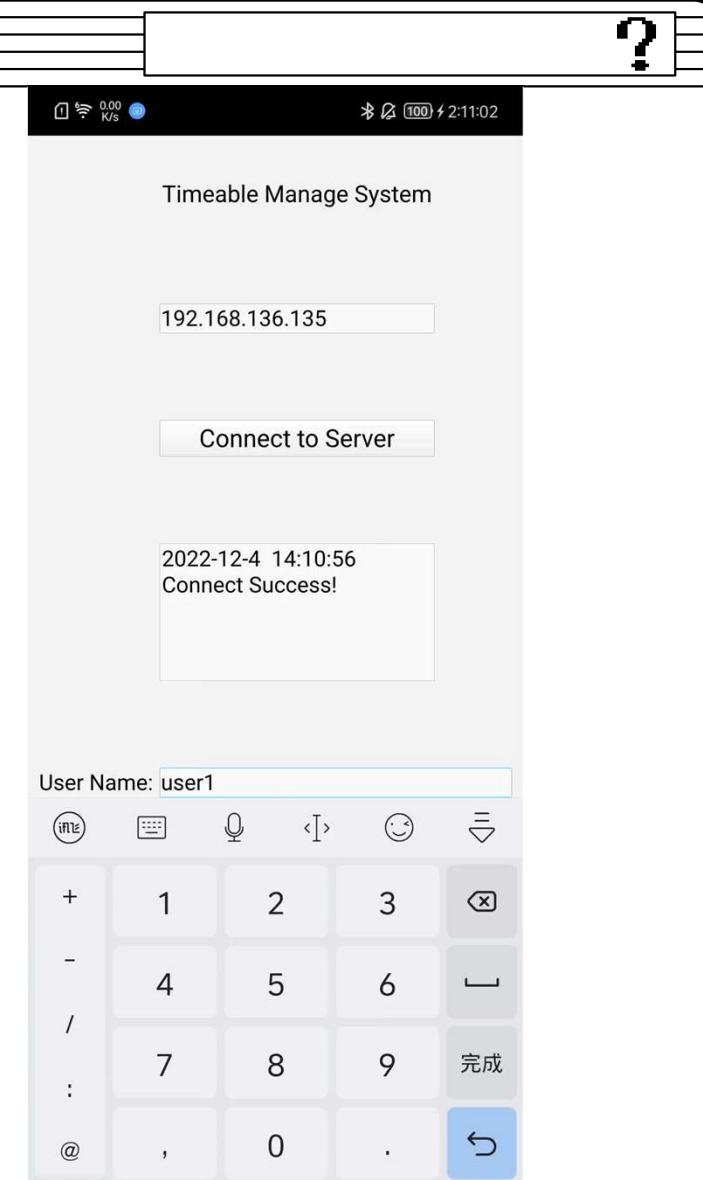
User input user name
If user is administrator,
then enter the admin
interface.



User input user name
If user is student, then
enter the course
searching interface.



User login





Student check timetable

A smartphone screen displaying a search interface. At the top is a black status bar with icons for signal strength, battery level, and time (2:21:41). Below the status bar is a white header with the text "Course Title" and a search button labeled "Search". The main area is a white space with a light gray border, intended for displaying search results.

Course Title Search

Exit

First search Computer,
then search Math

Student check timetable

If course can not
find , tell student.



Timeable Manage System

192.168.136.135

Connect to Server

2022-12-4 14:34:31
Connect Success!

User Name: user1

Password: 1234

Login

Exit



Student logout

If student push
“Exit” button, the
client will exit, and
the database will
upload this history.

162	user1	192.168.136.240	2022-12-4	15:21:26	6
163	user1	192.168.136.240	2022-12-4	15:32:57	10
164	user1	192.168.136.240	2022-12-4	15:37:9	5
165	admin1	192.168.136.240	2022-12-4	15:37:44	10

Course Title computer 2.08 K/s 100% 3:34:36

1793, TRA3123, Computer-Aided Translation , 1001, T4-301, Tue 11:00-11:50, CLC Teacher 17

1633, PRA4083, Human Computer Interaction , 1001, T4-302, Wed 9:00-9:50, Dr. Shawn TANG

1367, MAD4023, Computer Programming for Design , 1001, T3-409, Fri 14:00-14:50, Dr. William CHOW

1366, MAD4023, Computer Programming for Design , 1001, T3-501, Wed 10:00-11:50, Dr. Chenguang HU

1363, MAD3103, Computer Game Design , 1001, T3-707, Mon 9:00-11:50, Dr. Meng HE

842, TRA3123, Computer-Aided Translation , 1001, T4-409, Mon 10:00-11:50, CLC Teacher 17

734, PRA4083, Human Computer Interaction , 1001, T4-510, Mon 10:00-11:50, Ms. Yan-Yan JI

492, COMP4033, Computer Graphics , 1001, T5-705, Wed 13:00-13:50, Dr. Hao SHI

404, COMP4033, Computer Graphics , 1001, T5-605, Mon 10:00-11:50, Prof. Siu Kee WONG



22:08 1.1K/s 5G 66%



Administrator manages student

Timeable Manage System

192.168.136.135

Connect to Server

Connect Success! 2022-12-5
22:8:26

User Name: admin1

Password: 1234

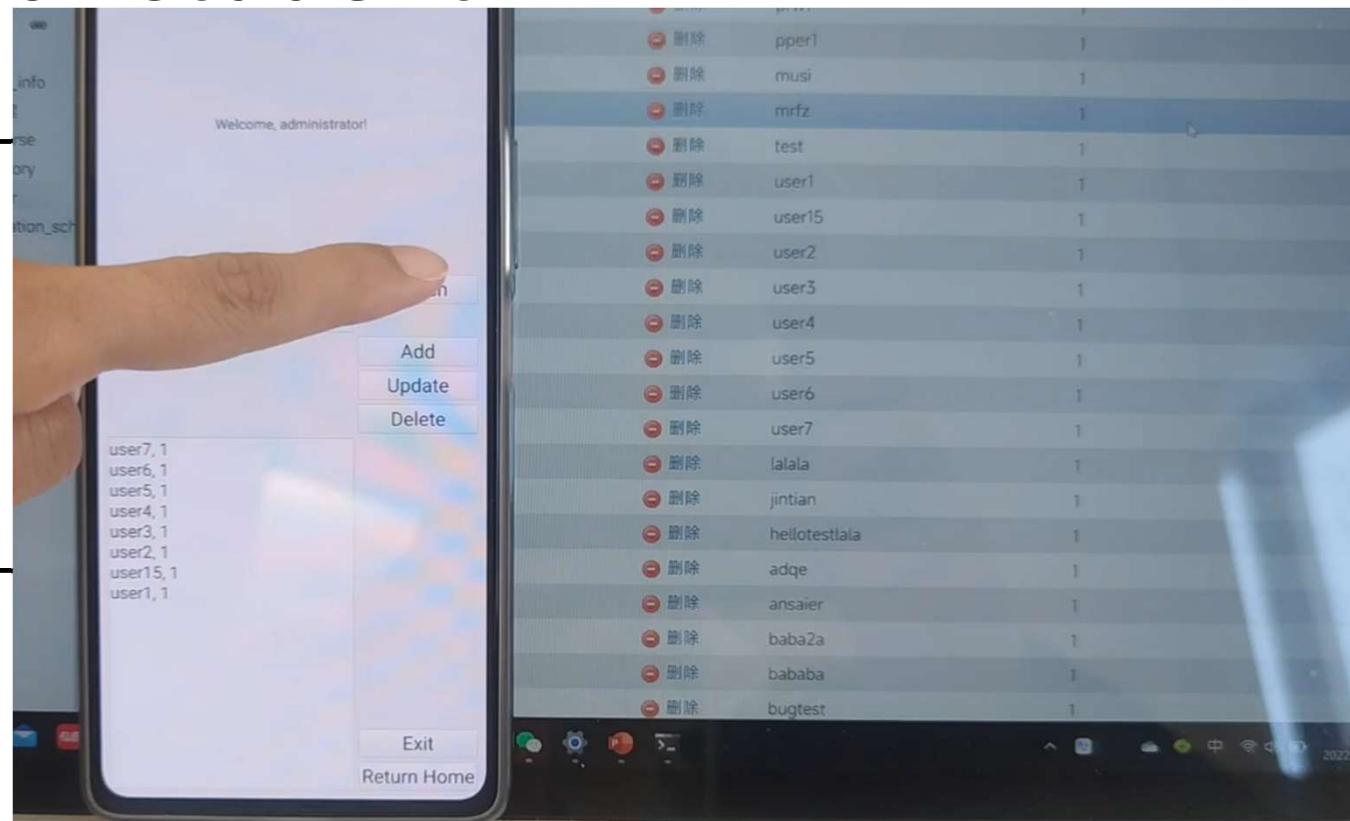
Login

Exit



Administrator search student

Input student name



0.08
K/s

100 3:42:01



Administrator add student

Input student name, and push “Add” button.



Please Input Student Name:

Search

Add

Update

Delete

Exit

Return Home

Administrator add student

Welcome, administrator!

If this student is already exist, add will fail.



Please Input Student Name:

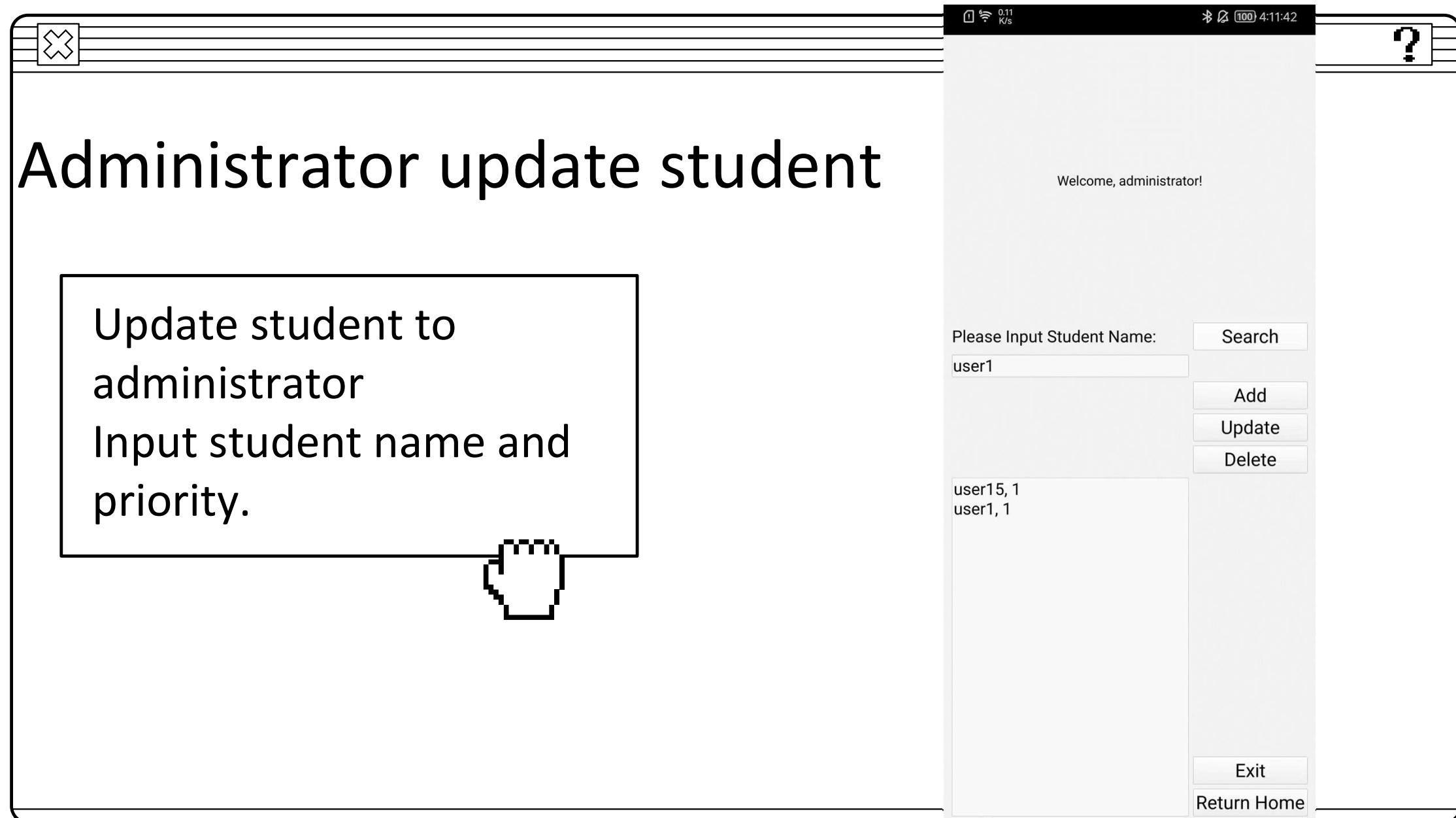
Search

Add

Update

Delete

Exit



Administrator update student

Update student to
administrator
Input student name and
priority.



Welcome, administrator!

Please Input Student Name:

user1

Search

Add

Update

Delete

user15, 1
user1, 1

Exit

Return Home

0.00
K/s

100 4:14:56



Administrator update student

Welcome, administrator!

If we can not find this student in database, update failed.



Please Input Student Name:

Search

Add

Update

Delete

user15, 1
user1, 2

The virtual keyboard includes standard letters (q-w, a-s, z), punctuation (., !, ?), numbers (123), and symbols (符, 中/英, 完成). It also features a backspace key (⌫) and a search/microphone icon (🔍).



Administrator delete student

Input student's name, and push delete button.

Welcome, administrator!

Please Input Student Name:

user7, 1
user6, 1
user5, 1
user4, 1
user3, 1
user2, 1
user15, 1
user1, 2

Administrator delete student

If this student
doesn't exist,
delete will fail.

Welcome, administrator!

Please Input Student Name:

Search

Add

Update

Delete

can not find this user



q w e r t y u i o p

a s d f g h j k l

↑ z x c v b n m ⌂

123 符 , . 中 英 完成



22:08



4.3K/s



5G



66%



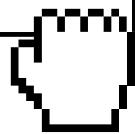
Administrator updates timetable

Please choose your operation:

[Student_info](#)[Timetable_info](#)

Update Course Timetable

Enter course mid, room,
and time to update the
timetable.



0.00
K/s 4:37:29

Update Timetable

course mid:

room information:

time information:

1

T5-401

Wed 12:00-12:50

Confirm

IMEI: Q ⌂ ☺ ⌄

q w e r t y u i o p

a s d f g h j k l

↑ z x c v b n m ⌈

123 符 , . 中英 完成



Update Course Timetable

If room or time crash,
update fail.



Update Timetable

course mid:

room information:

time information:

1

T5-309

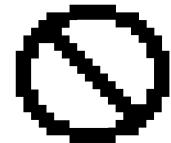
Thu 23:00-23:50

Confirm

This time or classroom
crash.

Exit





Thank You

彭容微2030026116, 钱景彬2030026117, 冯子骜2030026036