

# Introduction to Android TCP/IP Socket Server

CS 436 Software Development on Mobile

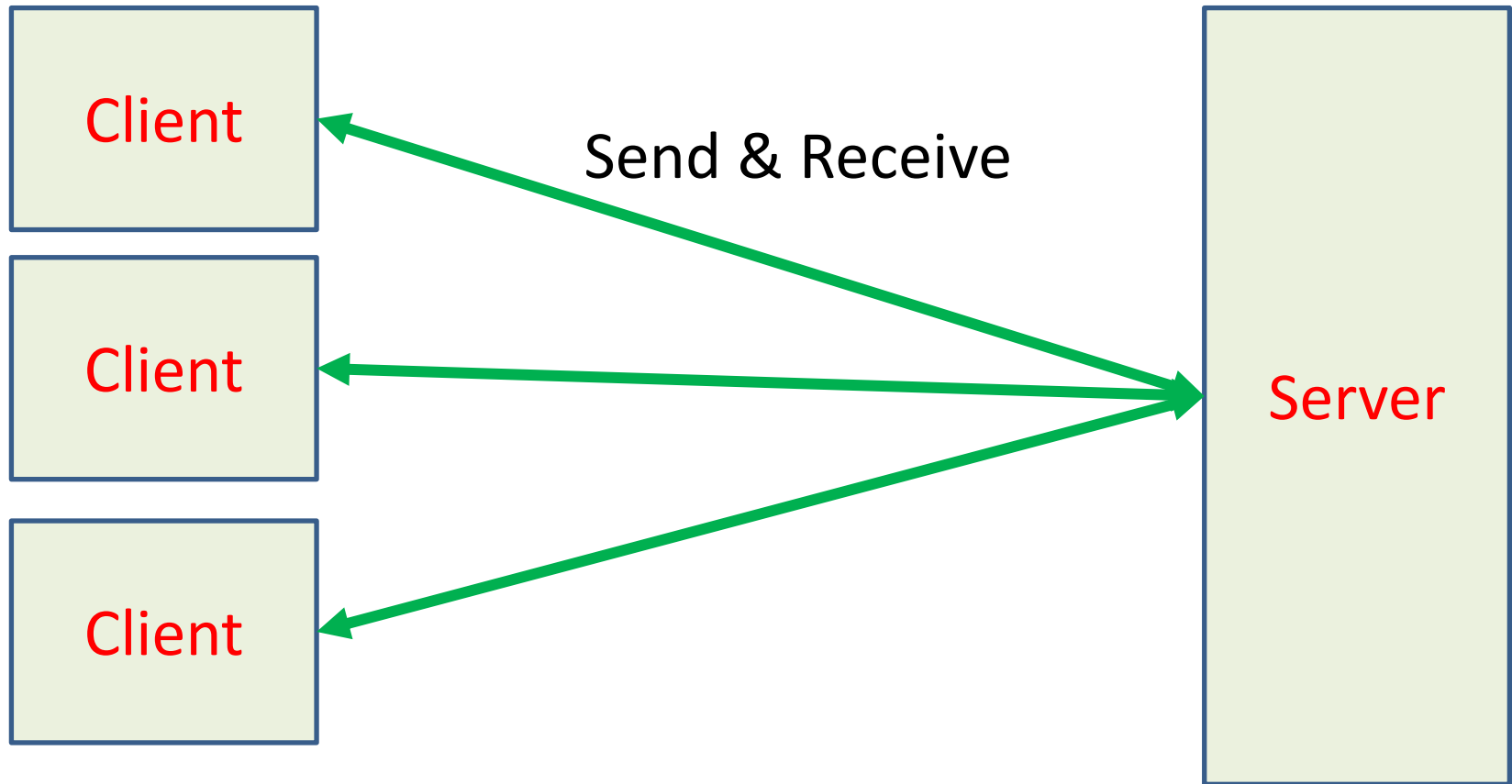
**Dr.Paween Khoenkaw**

Department of Computer Science  
Maejo University

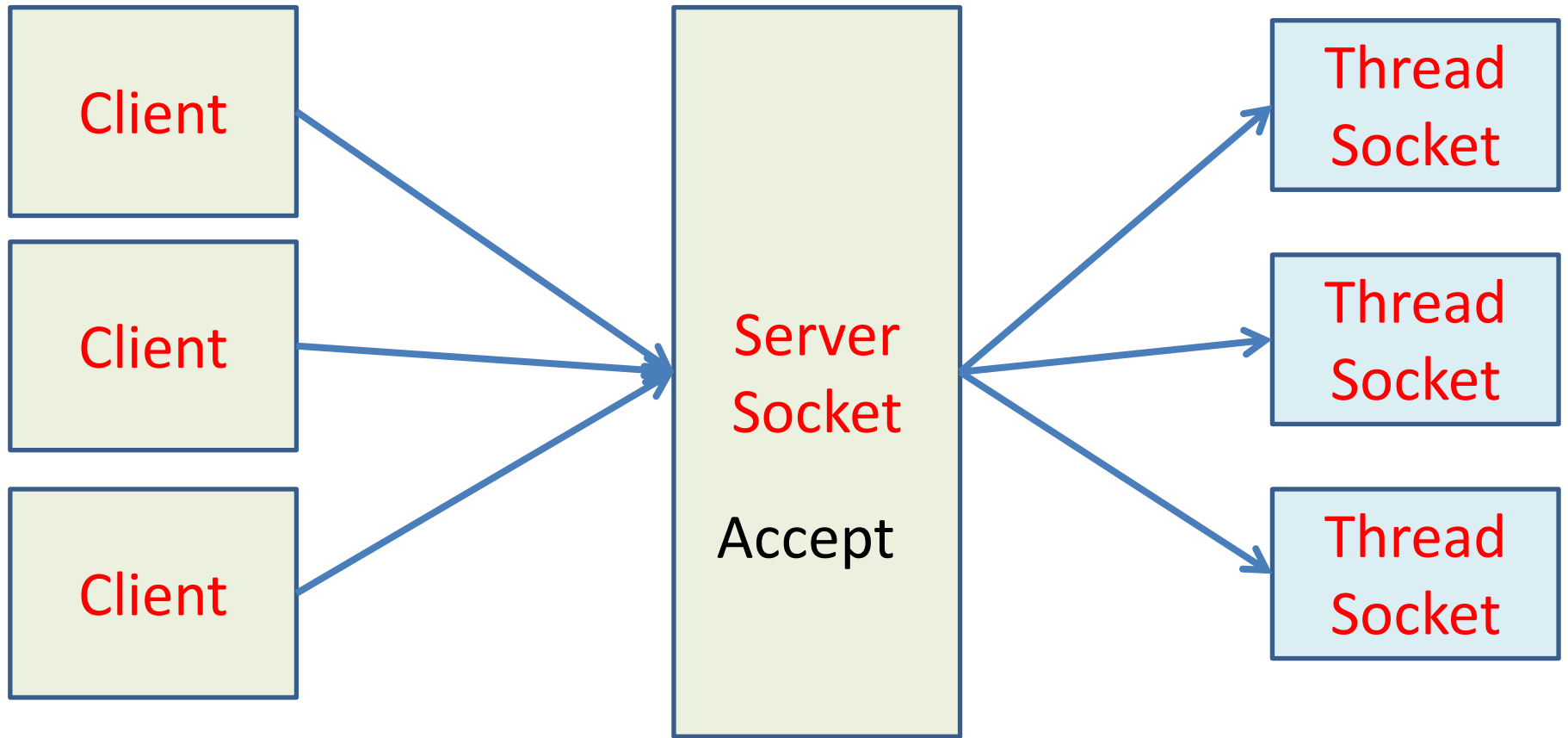


# Socket Server

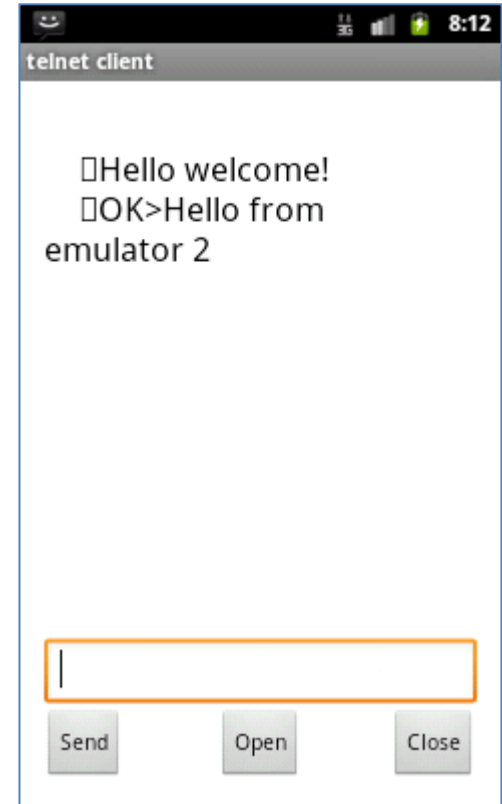
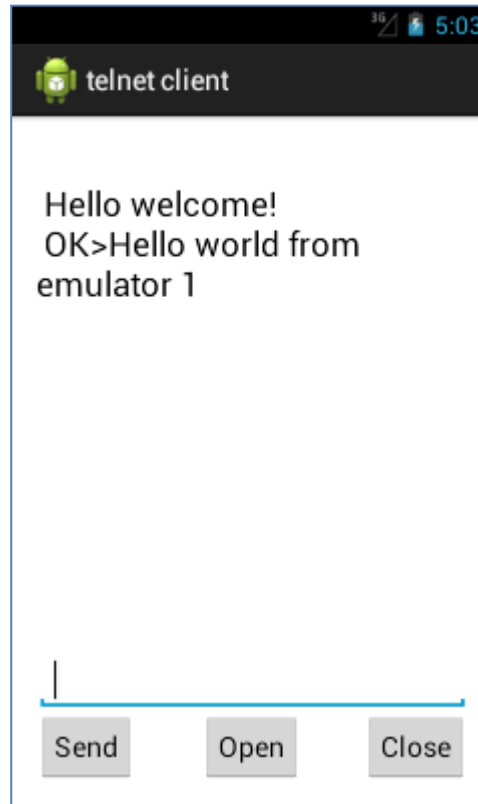
# Socket Server



# Socket Server

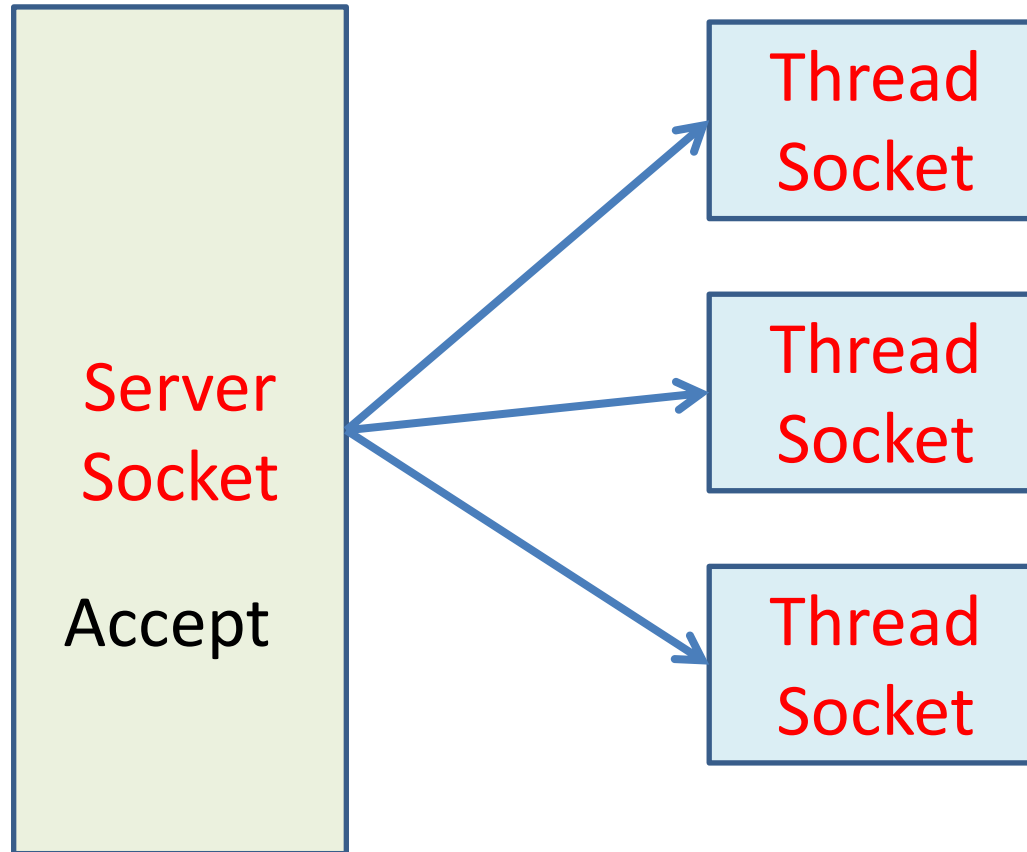


# Socket Server



Project: android\_communication\_tcpserver

# Socket Server



Class:ServerSocketThread  
Runnable interface

Class:AcceptClient  
Thread class

# Socket Server

```
public class ServerSocketThread implements Runnable {  
    @Override  
    public void run() {  
        try {  
            serverSocket = new ServerSocket(serverPort);  
            while (isListening) {  
                Socket client = serverSocket.accept();  
                AcceptClient obClient=new AcceptClient(client);  
            }  
            serverSocket.close();  
        } catch (IOException e) {  
        } catch (Exception e) {  
        }  
    }  
}
```

} Block Loop

ServerSocketThread

# Socket Server

```
class AcceptClient extends Thread
{ Socket ClientSocket;
  DataInputStream din;
  DataOutputStream dout;
  AcceptClient(Socket Csoc)
  { ClientSocket=Csoc;
    try {
din=new DataInputStream(ClientSocket.getInputStream());
  dout=new DataOutputStream(ClientSocket.getOutputStream());
  dout.writeUTF("Hello welcome!\n");
String msg=ClientSocket.getInetAddress().toString()+":"+ClientSocket.getPort()+" Just connected";
updater.setText(msg);h.post(updater);
  ClientSockets.add(ClientSocket);
  start();
    } catch (IOException e) {
    }
  }
  .....
```

AcceptClient (constructor)



# Socket Server

```
public void run() {
super.run();
while(true)
{
String msgFromClient=new String();
try {
msgFromClient=din.readLine();
... Check connection still alive ...
... Check client send "exit" command ...
String msg=ClientSocket.getInetAddress().toString()+":"+ClientSocket.getPort()+">"+ msgFromClient;
updater.setText(msg);h.post(updater);
dout.writeUTF("OK>"+msgFromClient+"\n");
} catch (IOException e) {
}
}
}
```

AcceptClient (Buffer Loop)

# Socket Server

```
if(msgFromClient==null)
{
String msg=ClientSocket.getInetAddress().toString()+":"+ClientSocket.getPort()+">" + "Connection
reset by peer";
ClientSocket.close();
updater.setText(msg);h.post(updater);
break;
}
if(msgFromClient.equals("exit"))
{
String msg=ClientSocket.getInetAddress().toString()+":"+ClientSocket.getPort()+">" + " Terminated!!";
dout.writeUTF("Bye :-)\n");
ClientSocket.close();
updater.setText(msg);h.post(updater);
break;
}
```

AcceptClient (Kill connection)

# Socket Server

Create ServerSocketThread instance and Show IP Address

```
final Thread fst=new Thread(new ServerSocketThread());  
textView1.setText(String.format("Server IP:%s Port:%d", NetUtil.getIPAddress(true)  
,serverPort));
```

Start Server

```
isListening=true;  
fst.start();
```

Kill Server

```
System.exit(0);
```

GUI

