

PROJECT REPORT:

I REALLY SPENT A LOT OF EFFORT AND TIME DOING THE PROJECT. IT TOOK ME ABOUT 3 MONTHS TO COMPLETE THIS PROJECT. FIRSTLY, I FOUND THE SOURCE OCTOBER OF THE PROJECT CLOSEST TO IT ON GITHUB, AND IN ADDITION, I REVIEWED ABOUT 200 300 PROGRAMS. GRADUALLY I STARTED WRITING THE PROGRAM. FIRST OF ALL, I DESIGNED THE INTERFACE OF THE PROGRAM. I'VE DONE THE BUTTON POSITIONING.I PUT THE LOGO OF THE MILITARY AND TURNED ON 5 WEBVIEW SCREENS SO THAT WE COULD WATCH 5 CAMERAS AT THE SAME TIME. (IF YOU ASK WHY THERE ARE 5 CAMERAS, THE REASON IS BECAUSE 5 CAMERAS ARE REQUIRED TO BE WATCHED AT THE SAME TIME IN THE PROJECT) I CREATED TEXTS UNDER THE WEB VIEWS AND SPECIFIED WHICH CAMERA IS FACING WHICH WAY (FOR EXAMPLE, STARBOARD MAIN MACHINE JPU-5), THEN I SWITCHED TO THE MAIN ACTIVITY SECTION AND STARTED BUTTON TRIGGERS. AFTER I DID THE TRIGGERING OF EACH BUTTON, I STARTED CALLING THE CODES VIA FTP. AFTER COMPLETING THE NECESSARY CODE OPERATIONS, I HAD TO DECODE 5 CAMERAS VIA SEQUENTIAL IP AND PRINT THE IMAGE AND GAVE THE COMMANDS BETWEEN 192.168.1.180 -185. BY THE WAY, THIS DEC DID NOT BOTHER MUCH, AS I WAS ABLE TO ASSIGN THE NECESSARY ROPES FROM THE CAMERA'S OWN ROPE MENU. AFTER COMPLETING THIS PROCESS, I RECEIVED CODES SUCH AS MSX MODE TRIGGERING, THERMAL MODE TRIGGERING FROM THE CAMERA'S OWN MENU. IT WAS EASY TO ACCESS THESE CODES, THANKS TO THE FACT THAT GOOGLE CHROME HAS THE ABILITY TO RIGHT-CLICK AND REVIEW. I HAD TO SOLVE THE ALARM SYSTEM OF THE CAMERAS AFTER I FINISHED THE BUTTON TRIGGER OPERATIONS. FOR THIS REASON, I AGAIN EXAMINED THE COMMANDS AND SOUNDS THAT THE CAMERA DOES NOT ALARM OVER ITS OWN IP ADDRESS, AND THEN ADAPTED THIS PROCESS TO MY OWN PROGRAM, WHICH IS HOW I MANAGED TO DO IT. FINALLY, THIS CAMERA HAS A LIGHT-ON FEATURE. I HAVE INTEGRATED THIS LIGHT-UP FEATURE INTO THE PROGRAM I AM WRITING AGAIN. AFTER COMPLETING MY LAST TESTS, I COMPILED IT AND TURNED IT INTO AN APK, AND IT TURNED OUT TO BE A VERY GOOD APPLICATION. I MADE THIS PROJECT TO THE COAST GUARD COMMAND AND THEY ARE STILL ACTIVELY USING IT NOW.

I leave the source code of my project below

CONTENT MAIN

<?xml version="1.0" encoding="utf-8"?>

<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"

xmlns:app="http://schemas.android.com/apk/res-auto"

xmlns:tools="http://schemas.android.com/tools"

android:id="@+id/MyLinear"

android:layout\_width="match\_parent"

android:layout\_height="match\_parent"

android:gravity="center\_horizontal"

android:orientation="vertical"

android:paddingLeft="8dp"

android:paddingTop="8dp"

android:paddingRight="8dp"

app:layout\_behavior="@string/appbar\_scrolling\_view\_behavior"

tools:context="com.project.shrey.ftptrial.com.project.TopSecretKeyIRFlir.ftptrial.MainActivity"

tools:showIn="@layout/activity\_main">

<Button

android:id="@+id/dirChooser"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:layout\_gravity="center"

android:text="Change Directory"

android:visibility="visible" />

<ImageView

android:id="@+id/imageView\_last"

android:layout\_width="wrap\_content"

android:layout\_height="329dp"

android:src="@drawable/header"

app:srcCompat="@drawable/header"

tools:srcCompat="@drawable/header" />

<WebView

android:id="@+id/WebViewSEL"

android:layout\_width="964dp"

android:layout\_height="329dp"

android:visibility="gone"></WebView>

<LinearLayout

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:layout\_marginBottom="5dp"

android:layout\_weight="1">

<Button

android:id="@+id/BTimageView"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:background="@color/colorText"

android:onClick="mBTimageView\_onClick"

android:text="AlarmOFF"

android:visibility="visible" />

<RelativeLayout

android:id="@+id/relativeLayout1"

android:layout\_width="match\_parent"

android:layout\_height="match\_parent"

android:gravity="center|center\_horizontal">

<ImageView

android:id="@+id/BTimageViewir\_1"

android:layout\_width="35dp"

android:layout\_height="35dp"

android:layout\_gravity="center"

android:layout\_marginLeft="280"

android:onClick="BTimageViewir\_tag\_onClick"

android:src="@drawable/ir\_btn\_1"

android:tag="1"

android:visibility="visible"

tools:src="@drawable/layer\_1" />

<ImageView

android:id="@+id/BTimageViewir\_2"

android:layout\_width="35dp"

android:layout\_height="35dp"

android:layout\_marginLeft="320"

android:onClick="BTimageViewir\_tag\_onClick"

android:src="@drawable/ir\_btn\_2"

android:tag="2"

android:visibility="visible"

tools:src="@drawable/layer\_2" />

<ImageView

android:id="@+id/BTimageViewir\_3"

android:layout\_width="35dp"

android:layout\_height="35dp"

android:layout\_marginLeft="360"

android:onClick="BTimageViewir\_tag\_onClick"

android:src="@drawable/ir\_btn\_3"

android:tag="3"

android:visibility="visible"

tools:src="@drawable/layer\_3" />

<ImageView

android:id="@+id/BTimageViewled"

android:layout\_width="35dp"

android:layout\_height="35dp"

android:layout\_marginLeft="405"

android:onClick="BTimageViewled\_onClick"

android:src="@drawable/led\_btn"

android:visibility="visible"

tools:src="@drawable/lamp\_on" />

</RelativeLayout>

<ImageView

android:id="@+id/BTimageViewir"

android:layout\_width="35dp"

android:layout\_height="35dp"

android:layout\_marginLeft="300dp"

android:layout\_marginTop="5dp"

android:onClick="BTimageViewir\_onClick"

android:src="@drawable/ir\_btn"

android:visibility="visible" />

</LinearLayout>

<LinearLayout

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:gravity="center"

android:layout\_weight="1">

<WebView

android:id="@+id/WebViewIR1"

android:layout\_width="380px"

android:layout\_height="292px"

android:layout\_marginStart="5px"

android:layout\_marginLeft="5px"

android:visibility="visible">

</WebView>

<WebView

android:id="@+id/WebViewIR2"

android:layout\_width="380px"

android:layout\_height="292px"

android:layout\_marginStart="5px"

android:layout\_marginLeft="5px"

android:visibility="visible">

</WebView>

<WebView

android:id="@+id/WebViewIR3"

android:layout\_width="380px"

android:layout\_height="292px"

android:layout\_marginStart="5px"

android:layout\_marginLeft="5px"

android:visibility="visible">

</WebView>

<WebView

android:id="@+id/WebViewIR4"

android:layout\_width="380px"

android:layout\_height="292px"

android:layout\_marginStart="5px"

android:layout\_marginLeft="5px"

android:visibility="visible">

</WebView>

<WebView

android:id="@+id/WebViewIR5"

android:layout\_width="380px"

android:layout\_height="292px"

android:layout\_marginStart="5px"

android:layout\_marginLeft="5px"

android:visibility="visible">

</WebView>

</LinearLayout>

<LinearLayout

android:layout\_width="match\_parent"

android:layout\_height="35dp"

android:layout\_weight="1"

android:gravity="center">

<Button

android:id="@+id/BTimageView11"

android:layout\_width="380px"

android:layout\_height="wrap\_content"

android:layout\_marginStart="5px"

android:layout\_marginLeft="5px"

android:background="@color/colorText"

android:onClick="opw"

android:tag="1"

android:text="Sancak Ana Makine"

android:visibility="visible" />

<Button

android:id="@+id/BTimageView12"

android:layout\_width="380px"

android:layout\_height="wrap\_content"

android:layout\_marginStart="5px"

android:layout\_marginLeft="5px"

android:background="@color/colorText"

android:onClick="opw"

android:tag="2"

android:text="İskele Ana Makine"

android:visibility="visible" />

<Button

android:id="@+id/BTimageView13"

android:layout\_width="380px"

android:layout\_height="wrap\_content"

android:layout\_marginStart="5px"

android:layout\_marginLeft="5px"

android:background="@color/colorText"

android:onClick="opw"

android:tag="3"

android:text="Baş Yardımcı"

android:visibility="visible" />

<Button

android:id="@+id/BTimageView14"

android:layout\_width="380px"

android:layout\_height="wrap\_content"

android:layout\_marginStart="5px"

android:layout\_marginLeft="5px"

android:background="@color/colorText"

android:onClick="opw"

android:tag="4"

android:text="Kıç Yardımcı"

android:visibility="visible" />

<Button

android:id="@+id/BTimageView15"

android:layout\_width="380px"

android:layout\_height="wrap\_content"

android:layout\_marginStart="5px"

android:layout\_marginLeft="5px"

android:background="@color/colorText"

android:onClick="opw"

android:tag="5"

android:text="JP5-Karosen"

android:visibility="visible" />

</LinearLayout>

<LinearLayout

android:id="@+id/addr1"

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:gravity="center"

android:visibility="visible">

<TextView

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:gravity="center"

android:paddingTop="6dp"

android:paddingBottom="6dp"

android:text="@string/sysApple"

android:textColor="@color/colorText"

android:textSize="14sp"

android:visibility="visible" />

<TextView

android:id="@+id/addriOS"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:gravity="center"

android:paddingTop="6dp"

android:paddingBottom="6dp"

android:text="@string/def\_addr\_ios"

android:textColor="#e8e8e8"

android:textSize="14sp"

android:visibility="visible" />

</LinearLayout>

<LinearLayout

android:id="@+id/addr2"

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:gravity="center"

android:visibility="visible">

<TextView

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:gravity="center"

android:paddingTop="6dp"

android:paddingBottom="6dp"

android:text="@string/sysOthers"

android:textColor="@color/colorText"

android:textSize="14sp" />

<TextView

android:id="@+id/addrReg"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:gravity="center"

android:paddingTop="6dp"

android:paddingBottom="6dp"

android:text="@string/def\_addr\_reg"

android:textColor="#e8e8e8"

android:textSize="14sp" />

</LinearLayout>

<RelativeLayout

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:gravity="center"

tools:visibility="invisible">

<TextView

android:id="@+id/userDisp"

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:gravity="left"

android:padding="10dp"

android:text="@string/defaultSetting"

android:textColor="@color/colorText"

android:textSize="18sp"

android:textStyle="bold"

android:visibility="invisible"

tools:visibility="invisible" />

<android.support.design.widget.TextInputLayout

android:id="@+id/userParent"

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:textColorHint="#a3a3a3">

<EditText

android:id="@+id/user"

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:backgroundTint="@color/editText"

android:inputType="text"

android:padding="15dp"

android:textColor="@color/editText"

android:textColorHint="@color/editTextCH"

android:visibility="invisible" />

</android.support.design.widget.TextInputLayout>

</RelativeLayout>

<RelativeLayout

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:gravity="center"

android:visibility="invisible">

<TextView

android:id="@+id/passwdDisp"

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:gravity="left"

android:padding="10dp"

android:text="@string/defaultSetting"

android:textColor="@color/colorText"

android:textSize="18sp"

android:textStyle="bold"

android:visibility="invisible" />

<android.support.design.widget.TextInputLayout

android:id="@+id/passwdParent"

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:textColorHint="#a3a3a3"

android:visibility="visible">

<EditText

android:id="@+id/passwd"

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:backgroundTint="@color/editText"

android:hint="@string/hintPassword"

android:inputType="textPassword"

android:padding="15sp"

android:textColor="@color/editText"

android:textColorHint="@color/editTextCH" />

</android.support.design.widget.TextInputLayout>

</RelativeLayout>

<Switch

android:id="@+id/togglePass"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:layout\_gravity="left"

android:layout\_margin="4dp"

android:gravity="center"

android:padding="4dp"

android:text="@string/passwordShow"

android:textColor="#e8e8e8"

android:textSize="15sp"

android:visibility="invisible" />

<TextView

android:id="@+id/dirAddress"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:layout\_margin="5dp"

android:text="@string/pathHomeExt"

android:textColor="#e8e8e8"

android:textSize="20sp"

android:visibility="invisible" />

<TextView

android:id="@+id/prompt"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:layout\_marginTop="6dp"

android:gravity="center"

android:padding="6dp"

android:text="@string/serverPrompt"

android:textColor="@color/colorText"

android:textSize="17sp"

android:visibility="invisible" />

</LinearLayout>

ACTIVITY MAIN:

<?xml version="1.0" encoding="utf-8"?>

<android.support.design.widget.CoordinatorLayout xmlns:android="http://schemas.android.com/apk/res/android"

xmlns:app="http://schemas.android.com/apk/res-auto"

xmlns:tools="http://schemas.android.com/tools"

android:layout\_width="match\_parent"

android:layout\_height="match\_parent"

android:background="@drawable/bg\_main"

tools:context="com.project.shrey.ftptrial.com.project.TopSecretKeyIRFlir.ftptrial.MainActivity">

<LinearLayout

android:layout\_width="match\_parent"

android:layout\_height="match\_parent"

android:orientation="vertical">

<android.support.design.widget.AppBarLayout

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:theme="@style/AppTheme.AppBarOverlay"

android:visibility="visible">

<android.support.v7.widget.Toolbar

android:id="@+id/toolbar"

android:layout\_width="match\_parent"

android:layout\_height="match\_parent"

android:background="@color/colorRed"

android:clickable="true"

android:focusable="true"

app:popupTheme="@style/AppTheme.PopupOverlay">

<ImageView

android:id="@+id/imageView112"

android:layout\_width="wrap\_content"

android:layout\_height="75dp"

android:adjustViewBounds="true"

android:scaleType="fitXY"

android:src="@drawable/header"

android:visibility="visible"

app:srcCompat="@drawable/header"

tools:srcCompat="@drawable/header" />

<ImageView

android:id="@+id/imageView2"

android:layout\_width="wrap\_content"

android:layout\_height="75dp"

android:adjustViewBounds="true"

android:scaleType="fitXY"

android:src="@drawable/ir"

android:visibility="visible"

app:srcCompat="@drawable/ir"

tools:srcCompat="@drawable/ir"

android:layout\_marginStart="15dp"

android:layout\_marginLeft="15dp" />

<ImageView

android:layout\_width="35dp"

android:layout\_height="35dp"

android:layout\_gravity="end"

android:layout\_marginEnd="15dp"

android:layout\_marginRight="15dp"

android:contentDescription="@string/ImgPower"

android:src="@drawable/power\_btn" />

</android.support.v7.widget.Toolbar>

</android.support.design.widget.AppBarLayout>

<android.support.v4.widget.NestedScrollView

android:layout\_width="match\_parent"

android:layout\_height="match\_parent">

<LinearLayout

android:layout\_width="match\_parent"

android:layout\_height="match\_parent"

android:orientation="vertical">

<include layout="@layout/content\_main" />

</LinearLayout>

</android.support.v4.widget.NestedScrollView>

</LinearLayout>

</android.support.design.widget.CoordinatorLayout>

MAIN ACTİVİTY:

package com.FHentertainment.dostool;

import android.app.ProgressDialog;

import android.content.Context;

import android.content.DialogInterface;

import android.content.Intent;

import android.content.res.Resources;

import android.media.MediaPlayer;

import android.net.Uri;

import android.net.wifi.WifiManager;

import android.os.Bundle;

import android.os.Environment;

import android.os.Handler;

import android.os.Looper;

import android.os.StrictMode;

import android.support.design.widget.TextInputLayout;

import android.support.p001v4.app.ActivityCompat;

import android.support.p001v4.content.ContextCompat;

import android.support.p001v4.content.res.ResourcesCompat;

import android.support.p004v7.app.AlertDialog;

import android.support.p004v7.app.AppCompatActivity;

import android.support.p004v7.widget.Toolbar;

import android.util.Log;

import android.util.TypedValue;

import android.view.View;

import android.view.WindowManager;

import android.webkit.WebView;

import android.webkit.WebViewClient;

import android.widget.Button;

import android.widget.CompoundButton;

import android.widget.EditText;

import android.widget.FrameLayout;

import android.widget.ImageView;

import android.widget.LinearLayout;

import android.widget.Switch;

import android.widget.TextView;

import java.io.BufferedReader;

import java.io.DataOutputStream;

import java.io.File;

import java.io.FileNotFoundException;

import java.io.IOException;

import java.io.InputStreamReader;

import java.io.PrintStream;

import java.net.CookieHandler;

import java.net.CookieManager;

import java.net.HttpURLConnection;

import java.net.URL;

import java.nio.charset.StandardCharsets;

import java.text.SimpleDateFormat;

import java.util.ArrayList;

import java.util.Date;

import java.util.HashMap;

import java.util.List;

import java.util.Locale;

import java.util.Map;

import net.rdrei.android.dirchooser.DirectoryChooserActivity;

import net.rdrei.android.dirchooser.DirectoryChooserConfig;

import org.apache.ftpserver.FtpServer;

import org.apache.ftpserver.FtpServerFactory;

import org.apache.ftpserver.ftplet.Authority;

import org.apache.ftpserver.ftplet.DefaultFtplet;

import org.apache.ftpserver.ftplet.FtpException;

import org.apache.ftpserver.ftplet.FtpReply;

import org.apache.ftpserver.ftplet.FtpRequest;

import org.apache.ftpserver.ftplet.FtpSession;

import org.apache.ftpserver.ftplet.Ftplet;

import org.apache.ftpserver.ftplet.FtpletContext;

import org.apache.ftpserver.ftplet.FtpletResult;

import org.apache.ftpserver.ftplet.UserManager;

import org.apache.ftpserver.listener.ListenerFactory;

import org.apache.ftpserver.usermanager.PropertiesUserManagerFactory;

import org.apache.ftpserver.usermanager.SaltedPasswordEncryptor;

import org.apache.ftpserver.usermanager.impl.BaseUser;

import org.apache.ftpserver.usermanager.impl.WritePermission;

import org.apache.log4j.varia.ExternallyRolledFileAppender;

import org.apache.mina.proxy.handlers.http.HttpProxyConstants;

public class MainActivity extends AppCompatActivity {

private static final String GET\_URL = "";

private static final String POST\_PARAMS = "";

private static final String POST\_URL = "";

private static final String USER\_AGENT = "Mozilla/5.0";

static String pass;

public int BTTAG = 0;

final int MY\_PERMISSIONS\_REQUEST = 2203;

final int REQUEST\_DIRECTORY = 2108;

String[] SCR\_ARR = {"192.168.1.181", "192.168.1.182", "192.168.1.183", "192.168.1.184", "192.168.1.185"};

int SCR\_CNT = 5;

public String client\_ip;

private Context context;

ListenerFactory factory = new ListenerFactory();

FtpServer finalServer;

boolean justStarted = true;

LinearLayout mAddr1;

LinearLayout mAddr2;

TextView mAddrReg;

TextView mAddriOS;

public int mBTimageViewIRLAST = 0;

public int mBTimageViewIR\_flag = 1;

public int mBTimageViewLED\_flag = 0;

ImageView mBTimageViewir;

ImageView mBTimageViewir\_1;

ImageView mBTimageViewir\_2;

ImageView mBTimageViewir\_3;

ImageView mBTimageViewled;

TextView mDirAddress;

Button mDirChooser;

LinearLayout mMyLinear;

EditText mPasswd;

TextView mPasswdDisp;

TextInputLayout mPasswdParent;

TextView mPrompt;

Switch mTogglePass;

EditText mUser;

TextView mUserDisp;

TextInputLayout mUserParent;

WebView mWebViewIR1;

WebView mWebViewIR2;

WebView mWebViewIR3;

WebView mWebViewIR4;

WebView mWebViewIR5;

public MediaPlayer mediaPlayer;

private ProgressDialog progress;

FtpServerFactory serverFactory = new FtpServerFactory();

Toolbar toolbar;

PropertiesUserManagerFactory userManagerFactory = new PropertiesUserManagerFactory();

/\* access modifiers changed from: protected \*/

public void onCreate(Bundle savedInstanceState) {

this.context = this;

super.onCreate(savedInstanceState);

setContentView((int) C0463R.layout.activity\_main);

Toolbar toolbar2 = (Toolbar) findViewById(C0463R.C0465id.toolbar);

this.toolbar = toolbar2;

setSupportActionBar(toolbar2);

getSupportActionBar().setDisplayShowTitleEnabled(false);

StrictMode.setThreadPolicy(new StrictMode.ThreadPolicy.Builder().permitAll().build());

if (ContextCompat.checkSelfPermission(this, "android.permission.WRITE\_EXTERNAL\_STORAGE") != 0) {

if (ActivityCompat.shouldShowRequestPermissionRationale(this, "android.permission.WRITE\_EXTERNAL\_STORAGE")) {

AlertDialog.Builder builder = new AlertDialog.Builder(this);

builder.setMessage((int) C0463R.string.dialog\_message).setTitle((int) C0463R.string.dialog\_title);

builder.setPositiveButton((CharSequence) ExternallyRolledFileAppender.f72OK, (DialogInterface.OnClickListener) new DialogInterface.OnClickListener() {

public final void onClick(DialogInterface dialogInterface, int i) {

MainActivity.this.lambda$onCreate$0$MainActivity(dialogInterface, i);

}

});

builder.show();

} else {

ActivityCompat.requestPermissions(this, new String[]{"android.permission.WRITE\_EXTERNAL\_STORAGE"}, 2203);

}

}

CookieHandler.setDefault(new CookieManager());

this.mTogglePass = (Switch) findViewById(C0463R.C0465id.togglePass);

this.mBTimageViewir\_1 = (ImageView) findViewById(C0463R.C0465id.BTimageViewir\_1);

this.mBTimageViewir\_2 = (ImageView) findViewById(C0463R.C0465id.BTimageViewir\_2);

this.mBTimageViewir\_3 = (ImageView) findViewById(C0463R.C0465id.BTimageViewir\_3);

this.mBTimageViewled = (ImageView) findViewById(C0463R.C0465id.BTimageViewled);

this.mTogglePass.setEnabled(false);

this.mTogglePass.setChecked(false);

this.mUserParent = (TextInputLayout) findViewById(C0463R.C0465id.userParent);

this.mUser = (EditText) findViewById(C0463R.C0465id.user);

this.mUserDisp = (TextView) findViewById(C0463R.C0465id.userDisp);

this.mPasswd = (EditText) findViewById(C0463R.C0465id.passwd);

this.mPasswdDisp = (TextView) findViewById(C0463R.C0465id.passwdDisp);

this.mPasswdParent = (TextInputLayout) findViewById(C0463R.C0465id.passwdParent);

this.mPrompt = (TextView) findViewById(C0463R.C0465id.prompt);

this.mMyLinear = (LinearLayout) findViewById(C0463R.C0465id.MyLinear);

this.mAddrReg = (TextView) findViewById(C0463R.C0465id.addrReg);

this.mAddr2 = (LinearLayout) findViewById(C0463R.C0465id.addr2);

this.mAddrReg.setText(String.format("ftp://%s:2121", new Object[]{wifiIpAddress(this)}));

this.mAddriOS = (TextView) findViewById(C0463R.C0465id.addriOS);

this.mAddr1 = (LinearLayout) findViewById(C0463R.C0465id.addr1);

this.mAddriOS.setText(String.format("ftp://ftp:ftp@%s:2121", new Object[]{wifiIpAddress(this)}));

this.mDirAddress = (TextView) findViewById(C0463R.C0465id.dirAddress);

this.mDirChooser = (Button) findViewById(C0463R.C0465id.dirChooser);

this.mediaPlayer = MediaPlayer.create(this, C0463R.raw.alarm);

if (this.SCR\_CNT >= 1) {

this.mWebViewIR1 = (WebView) findViewById(C0463R.C0465id.WebViewIR1);

}

if (this.SCR\_CNT >= 2) {

this.mWebViewIR2 = (WebView) findViewById(C0463R.C0465id.WebViewIR2);

}

if (this.SCR\_CNT >= 3) {

this.mWebViewIR3 = (WebView) findViewById(C0463R.C0465id.WebViewIR3);

}

if (this.SCR\_CNT >= 4) {

this.mWebViewIR4 = (WebView) findViewById(C0463R.C0465id.WebViewIR4);

}

if (this.SCR\_CNT >= 5) {

this.mWebViewIR5 = (WebView) findViewById(C0463R.C0465id.WebViewIR5);

}

if (this.SCR\_CNT >= 1) {

this.mWebViewIR1.getSettings().setJavaScriptEnabled(true);

this.mWebViewIR1.setWebViewClient(new WebViewClient() {

public void onPageFinished(WebView view, String url) {

super.onPageFinished(view, url);

WebView webView = MainActivity.this.mWebViewIR1;

webView.loadUrl("http://" + MainActivity.this.SCR\_ARR[0] + "/liveonly");

MainActivity.this.mWebViewIR1.setVisibility(0);

MainActivity.this.mWebViewIR1.setVisibility(8);

}

});

this.mWebViewIR1.setInitialScale(67);

this.mWebViewIR1.setVisibility(4);

WebView webView = this.mWebViewIR1;

webView.loadUrl("http://" + this.SCR\_ARR[0] + "/liveonly/admin:admin");

this.mWebViewIR1.setVisibility(8);

}

if (this.SCR\_CNT >= 2) {

this.mWebViewIR2.setInitialScale(67);

this.mWebViewIR2.getSettings().setJavaScriptEnabled(true);

this.mWebViewIR2.setWebViewClient(new WebViewClient() {

public void onPageFinished(WebView view, String url) {

super.onPageFinished(view, url);

WebView webView = MainActivity.this.mWebViewIR2;

webView.loadUrl("http://" + MainActivity.this.SCR\_ARR[1] + "/liveonly");

MainActivity.this.mWebViewIR2.setVisibility(0);

MainActivity.this.mWebViewIR2.setVisibility(8);

}

});

this.mWebViewIR2.setVisibility(4);

WebView webView2 = this.mWebViewIR2;

webView2.loadUrl("http://" + this.SCR\_ARR[1] + "/liveonly/admin:admin");

this.mWebViewIR2.setVisibility(8);

}

if (this.SCR\_CNT >= 3) {

this.mWebViewIR3.setInitialScale(67);

this.mWebViewIR3.getSettings().setJavaScriptEnabled(true);

this.mWebViewIR3.setWebViewClient(new WebViewClient() {

public void onPageFinished(WebView view, String url) {

super.onPageFinished(view, url);

WebView webView = MainActivity.this.mWebViewIR3;

webView.loadUrl("http://" + MainActivity.this.SCR\_ARR[2] + "/liveonly");

MainActivity.this.mWebViewIR3.setVisibility(0);

MainActivity.this.mWebViewIR3.setVisibility(8);

}

});

this.mWebViewIR3.setVisibility(4);

WebView webView3 = this.mWebViewIR3;

webView3.loadUrl("http://" + this.SCR\_ARR[2] + "/liveonly/admin:admin");

this.mWebViewIR3.setVisibility(8);

}

if (this.SCR\_CNT >= 4) {

this.mWebViewIR4.setInitialScale(67);

this.mWebViewIR4.getSettings().setJavaScriptEnabled(true);

this.mWebViewIR4.setWebViewClient(new WebViewClient() {

public void onPageFinished(WebView view, String url) {

super.onPageFinished(view, url);

WebView webView = MainActivity.this.mWebViewIR4;

webView.loadUrl("http://" + MainActivity.this.SCR\_ARR[3] + "/liveonly");

MainActivity.this.mWebViewIR4.setVisibility(0);

MainActivity.this.mWebViewIR4.setVisibility(8);

}

});

this.mWebViewIR4.setVisibility(4);

WebView webView4 = this.mWebViewIR4;

webView4.loadUrl("http://" + this.SCR\_ARR[3] + "/liveonly/admin:admin");

this.mWebViewIR4.setVisibility(8);

}

if (this.SCR\_CNT >= 5) {

this.mWebViewIR5.setInitialScale(67);

this.mWebViewIR5.getSettings().setJavaScriptEnabled(true);

this.mWebViewIR5.setWebViewClient(new WebViewClient() {

public void onPageFinished(WebView view, String url) {

super.onPageFinished(view, url);

WebView webView = MainActivity.this.mWebViewIR5;

webView.loadUrl("http://" + MainActivity.this.SCR\_ARR[4] + "/liveonly");

MainActivity.this.mWebViewIR5.setVisibility(0);

MainActivity.this.mWebViewIR5.setVisibility(8);

}

});

this.mWebViewIR5.setVisibility(4);

WebView webView5 = this.mWebViewIR5;

webView5.loadUrl("http://" + this.SCR\_ARR[4] + "/liveonly/admin:admin");

this.mWebViewIR5.setVisibility(8);

}

this.mDirChooser.setOnClickListener(new View.OnClickListener() {

public final void onClick(View view) {

MainActivity.this.lambda$onCreate$1$MainActivity(view);

}

});

this.finalServer = this.serverFactory.createServer();

this.toolbar.setOnClickListener(new View.OnClickListener() {

public final void onClick(View view) {

MainActivity.this.lambda$onCreate$3$MainActivity(view);

}

});

this.mTogglePass.setOnCheckedChangeListener(new CompoundButton.OnCheckedChangeListener() {

public final void onCheckedChanged(CompoundButton compoundButton, boolean z) {

MainActivity.this.lambda$onCreate$4$MainActivity(compoundButton, z);

}

});

getWindow().setSoftInputMode(16);

}

public /\* synthetic \*/ void lambda$onCreate$0$MainActivity(DialogInterface dialog, int id) {

dialog.dismiss();

this.justStarted = false;

ActivityCompat.requestPermissions(this, new String[]{"android.permission.WRITE\_EXTERNAL\_STORAGE"}, 2203);

}

public /\* synthetic \*/ void lambda$onCreate$1$MainActivity(View view) {

Intent chooserIntent = new Intent(this, DirectoryChooserActivity.class);

chooserIntent.putExtra(DirectoryChooserActivity.EXTRA\_CONFIG, DirectoryChooserConfig.builder().newDirectoryName("New Folder").allowNewDirectoryNameModification(true).build());

startActivityForResult(chooserIntent, 2108);

}

public /\* synthetic \*/ void lambda$onCreate$3$MainActivity(View view) {

if (checkWifiOnAndConnected(this)) {

serverControl();

return;

}

AlertDialog.Builder builder = new AlertDialog.Builder(this);

builder.setMessage((CharSequence) "wifiMgr.isWifiEnabled()").setTitle((CharSequence) "aaaaa");

builder.setMessage((int) C0463R.string.dialog\_wifi\_message).setTitle((int) C0463R.string.dialog\_wifi\_title);

builder.setPositiveButton((CharSequence) "OKRRR", (DialogInterface.OnClickListener) $$Lambda$MainActivity$26g07Bs5zFcLKgyfdu4ktpJ6nBM.INSTANCE);

builder.show();

}

public /\* synthetic \*/ void lambda$onCreate$4$MainActivity(CompoundButton compoundButton, boolean b) {

if (b) {

this.mPasswdDisp.setText(String.format("Password: %s", new Object[]{pass}));

return;

}

StringBuilder strB = new StringBuilder("Password: ");

for (int i = 0; i < pass.length(); i++) {

strB.append('\*');

}

this.mPasswdDisp.setText(strB.toString());

}

public void onRequestPermissionsResult(int requestCode, String[] permissions, int[] grantResults) {

if (requestCode == 2203) {

if (grantResults.length <= 0 || grantResults[0] != 0) {

AlertDialog.Builder builder = new AlertDialog.Builder(this);

builder.setMessage((int) C0463R.string.dialog\_message\_exit).setTitle((int) C0463R.string.dialog\_title);

builder.setPositiveButton((CharSequence) ExternallyRolledFileAppender.f72OK, (DialogInterface.OnClickListener) new DialogInterface.OnClickListener() {

public final void onClick(DialogInterface dialogInterface, int i) {

MainActivity.this.lambda$onRequestPermissionsResult$5$MainActivity(dialogInterface, i);

}

});

builder.show();

}

}

}

public /\* synthetic \*/ void lambda$onRequestPermissionsResult$5$MainActivity(DialogInterface dialog, int id) {

dialog.dismiss();

finish();

}

/\* access modifiers changed from: protected \*/

public void onActivityResult(int requestCode, int resultCode, Intent data) {

super.onActivityResult(requestCode, resultCode, data);

if (requestCode == 2108 && resultCode == 1) {

this.mDirAddress.setText(data.getStringExtra(DirectoryChooserActivity.RESULT\_SELECTED\_DIR));

}

}

/\* access modifiers changed from: protected \*/

public void onDestroy() {

try {

this.finalServer.stop();

} catch (Exception e) {

Log.e("Server Close Error", e.getCause().toString());

}

super.onDestroy();

}

private void setupStart(String username, String password, String subLoc) throws FileNotFoundException {

this.factory.setPort(2121);

this.serverFactory.addListener("default", this.factory.createListener());

File files = new File(Environment.getExternalStorageDirectory().getPath() + "/users.properties");

if (!files.exists()) {

try {

files.createNewFile();

} catch (IOException e) {

e.printStackTrace();

}

}

this.userManagerFactory.setFile(files);

this.userManagerFactory.setPasswordEncryptor(new SaltedPasswordEncryptor());

UserManager um = this.userManagerFactory.createUserManager();

BaseUser user = new BaseUser();

user.setName(username);

user.setPassword(password);

user.setHomeDirectory(Environment.getExternalStorageDirectory().getPath() + "/" + subLoc);

List<Authority> auths = new ArrayList<>();

auths.add(new WritePermission());

user.setAuthorities(auths);

try {

um.save(user);

} catch (FtpException e1) {

e1.printStackTrace();

}

this.serverFactory.setUserManager(um);

Map<String, Ftplet> m = new HashMap<>();

m.put("miaFtplet", new Ftplet() {

public void init(FtpletContext ftpletContext) throws FtpException {

}

public void destroy() {

}

public FtpletResult beforeCommand(FtpSession session, FtpRequest request) throws FtpException, IOException {

return FtpletResult.DEFAULT;

}

public FtpletResult afterCommand(FtpSession session, FtpRequest request, FtpReply reply) throws FtpException, IOException {

PrintStream printStream = System.out;

printStream.println("Ending uploading.... " + request.getCommand() + request.getArgument() + request.getRequestLine());

PrintStream printStream2 = System.out;

printStream2.println(": " + session.getUserArgument() + session.toString());

return FtpletResult.DEFAULT;

}

public FtpletResult onConnect(FtpSession session) throws FtpException, IOException {

return FtpletResult.DEFAULT;

}

public FtpletResult onDisconnect(FtpSession session) throws FtpException, IOException {

return FtpletResult.DEFAULT;

}

});

this.serverFactory.setFtplets(m);

HashMap<String, Ftplet> ftplets = new HashMap<>();

ftplets.put("1", new TraceFtp());

this.serverFactory.setFtplets(ftplets);

}

private String wifiIpAddress(Context context2) {

return Utils.getIPAddress(true);

}

private boolean checkWifiOnAndConnected(Context context2) {

WifiManager wifiManager = (WifiManager) context2.getApplicationContext().getSystemService("wifi");

return true;

}

public void onBackPressed() {

this.finalServer.stop();

findViewById(C0463R.C0465id.toolbar).setEnabled(false);

this.toolbar.setBackgroundColor(ResourcesCompat.getColor(getResources(), C0463R.color.colorRed, (Resources.Theme) null));

super.onBackPressed();

}

/\* access modifiers changed from: package-private \*/

public void startWEB() {

String currentDate = new SimpleDateFormat("yyyy-MM-dd", Locale.getDefault()).format(new Date());

if (currentDate.compareTo("2022-02-01") == 1) {

showExpire();

}

if (currentDate.compareTo("2022-02-01") == -1) {

if (this.SCR\_CNT >= 1) {

this.mWebViewIR1 = (WebView) findViewById(C0463R.C0465id.WebViewIR1);

}

if (this.SCR\_CNT >= 2) {

this.mWebViewIR2 = (WebView) findViewById(C0463R.C0465id.WebViewIR2);

}

if (this.SCR\_CNT >= 3) {

this.mWebViewIR3 = (WebView) findViewById(C0463R.C0465id.WebViewIR3);

}

if (this.SCR\_CNT >= 4) {

this.mWebViewIR4 = (WebView) findViewById(C0463R.C0465id.WebViewIR4);

}

if (this.SCR\_CNT >= 5) {

this.mWebViewIR5 = (WebView) findViewById(C0463R.C0465id.WebViewIR5);

}

if (this.SCR\_CNT >= 1) {

this.mWebViewIR1.getSettings().setJavaScriptEnabled(true);

this.mWebViewIR1.setWebViewClient(new WebViewClient() {

public void onPageFinished(WebView view, String url) {

super.onPageFinished(view, url);

WebView webView = MainActivity.this.mWebViewIR1;

webView.loadUrl("http://" + MainActivity.this.SCR\_ARR[0] + "/liveonly");

}

});

this.mWebViewIR1.loadUrl("http://" + this.SCR\_ARR[0] + "/liveonly/admin:admin");

}

if (this.SCR\_CNT >= 2) {

this.mWebViewIR2.getSettings().setJavaScriptEnabled(true);

this.mWebViewIR2.setWebViewClient(new WebViewClient() {

public void onPageFinished(WebView view, String url) {

super.onPageFinished(view, url);

WebView webView = MainActivity.this.mWebViewIR2;

webView.loadUrl("http://" + MainActivity.this.SCR\_ARR[1] + "/liveonly");

}

});

this.mWebViewIR2.loadUrl("http://" + this.SCR\_ARR[1] + "/liveonly/admin:admin");

}

if (this.SCR\_CNT >= 3) {

this.mWebViewIR3.getSettings().setJavaScriptEnabled(true);

this.mWebViewIR3.setWebViewClient(new WebViewClient() {

public void onPageFinished(WebView view, String url) {

super.onPageFinished(view, url);

WebView webView = MainActivity.this.mWebViewIR3;

webView.loadUrl("http://" + MainActivity.this.SCR\_ARR[2] + "/liveonly");

}

});

this.mWebViewIR3.loadUrl("http://" + this.SCR\_ARR[2] + "/liveonly/admin:admin");

}

if (this.SCR\_CNT >= 4) {

this.mWebViewIR4.getSettings().setJavaScriptEnabled(true);

this.mWebViewIR4.setWebViewClient(new WebViewClient() {

public void onPageFinished(WebView view, String url) {

super.onPageFinished(view, url);

WebView webView = MainActivity.this.mWebViewIR4;

webView.loadUrl("http://" + MainActivity.this.SCR\_ARR[3] + "/liveonly");

}

});

this.mWebViewIR4.loadUrl("http://" + this.SCR\_ARR[3] + "/liveonly/admin:admin");

}

if (this.SCR\_CNT >= 5) {

this.mWebViewIR5.getSettings().setJavaScriptEnabled(true);

this.mWebViewIR5.setWebViewClient(new WebViewClient() {

public void onPageFinished(WebView view, String url) {

super.onPageFinished(view, url);

WebView webView = MainActivity.this.mWebViewIR5;

webView.loadUrl("http://" + MainActivity.this.SCR\_ARR[4] + "/liveonly");

}

});

this.mWebViewIR5.loadUrl("http://" + this.SCR\_ARR[4] + "/liveonly/admin:admin");

}

if (this.SCR\_CNT >= 1) {

this.mWebViewIR1.setVisibility(0);

}

if (this.SCR\_CNT >= 2) {

this.mWebViewIR2.setVisibility(0);

}

if (this.SCR\_CNT >= 3) {

this.mWebViewIR3.setVisibility(0);

}

if (this.SCR\_CNT >= 4) {

this.mWebViewIR4.setVisibility(0);

}

if (this.SCR\_CNT >= 5) {

this.mWebViewIR5.setVisibility(0);

}

findViewById(C0463R.C0465id.BTimageView11).setVisibility(0);

findViewById(C0463R.C0465id.BTimageView12).setVisibility(0);

findViewById(C0463R.C0465id.BTimageView13).setVisibility(0);

findViewById(C0463R.C0465id.BTimageView14).setVisibility(0);

findViewById(C0463R.C0465id.BTimageView15).setVisibility(0);

for (int k = 0; k <= this.SCR\_CNT - 1; k++) {

String ff = "http://" + this.SCR\_ARR[k] + "/login/dologin";

String don = "";

try {

don = sendPOST(ff, "user\_name=admin&user\_password=admin");

Log.e("LOGIN", ff + don);

} catch (Exception ex) {

Log.e("LOGIN", "hata" + ff + ex.toString());

}

try {

ff = "http://" + this.SCR\_ARR[k] + "/res.php";

don = sendPOST(ff, "action=set&resource=.system.vcam.torch&value=false");

Log.e("LAMPON", ff + don);

} catch (Exception ex2) {

Log.e("LAMPON", "hata" + ff + don + ex2.toString());

}

try {

String ff2 = "http://" + this.SCR\_ARR[k] + "/home/setviewmode/IR";

don = sendGET(ff2);

Log.e("IRBUTTON", ff2 + don);

} catch (Exception e) {

Log.e("IRBUTTON", "hata" + don);

}

}

this.mBTimageViewir\_1.setVisibility(0);

this.mBTimageViewir\_2.setVisibility(0);

this.mBTimageViewir\_3.setVisibility(0);

this.mBTimageViewled.setVisibility(0);

}

}

/\* access modifiers changed from: package-private \*/

public void serverControl() {

String user;

if (this.finalServer.isStopped()) {

this.mUser.setEnabled(false);

this.mPasswd.setEnabled(false);

this.mDirChooser.setEnabled(false);

String user2 = this.mUser.getText().toString();

String passwd = this.mPasswd.getText().toString();

if (user2.isEmpty()) {

user = "ftp";

} else {

user = user2;

}

if (passwd.isEmpty()) {

passwd = "ftp";

}

String subLoc = this.mDirAddress.getText().toString().substring(20);

pass = passwd;

StringBuilder strB = new StringBuilder("Password: ");

for (int i = 0; i < pass.length(); i++) {

strB.append('\*');

}

this.mPasswdDisp.setText(strB.toString());

this.mUserDisp.setText(String.format("Username: %s", new Object[]{user}));

try {

setupStart(user, passwd, subLoc);

} catch (FileNotFoundException e) {

FileNotFoundException fileNotFoundException = e;

if (ActivityCompat.shouldShowRequestPermissionRationale(this, "android.permission.WRITE\_EXTERNAL\_STORAGE")) {

AlertDialog.Builder builder = new AlertDialog.Builder(this);

builder.setMessage((int) C0463R.string.dialog\_message\_error).setTitle((int) C0463R.string.dialog\_title);

builder.setPositiveButton((CharSequence) ExternallyRolledFileAppender.f72OK, (DialogInterface.OnClickListener) new DialogInterface.OnClickListener() {

public final void onClick(DialogInterface dialogInterface, int i) {

MainActivity.this.lambda$serverControl$6$MainActivity(dialogInterface, i);

}

});

builder.show();

} else {

ActivityCompat.requestPermissions(this, new String[]{"android.permission.WRITE\_EXTERNAL\_STORAGE"}, 2203);

}

}

try {

this.finalServer.start();

this.mAddrReg.setText(String.format("ftp://%s:2121", new Object[]{wifiIpAddress(this)}));

this.mAddriOS.setText(String.format("ftp://%s:%s@%s:2121", new Object[]{user, passwd, wifiIpAddress(this)}));

} catch (FtpException e2) {

e2.printStackTrace();

}

this.toolbar.setBackgroundColor(ResourcesCompat.getColor(getResources(), C0463R.color.colorGreen, (Resources.Theme) null));

final ProgressDialog Loaddialog = ProgressDialog.show(this, "", "Loading. Please wait...", true);

Handler handler = new Handler(Looper.getMainLooper());

handler.postDelayed(new Runnable() {

public void run() {

MainActivity.this.startWEB();

}

}, 100);

handler.postDelayed(new Runnable() {

public void run() {

Loaddialog.dismiss();

}

}, 500);

this.mPrompt.setVisibility(0);

this.mAddr1.setVisibility(0);

this.mAddr2.setVisibility(0);

this.mTogglePass.setEnabled(true);

this.mMyLinear.scrollTo(0, 0);

} else if (this.finalServer.isSuspended()) {

this.finalServer.resume();

this.toolbar.setBackgroundColor(ResourcesCompat.getColor(getResources(), C0463R.color.colorGreen, (Resources.Theme) null));

final ProgressDialog Loaddialog2 = ProgressDialog.show(this, "", "Loading. Please wait...", true);

Handler handler2 = new Handler(Looper.getMainLooper());

handler2.postDelayed(new Runnable() {

public void run() {

MainActivity.this.startWEB();

}

}, 100);

handler2.postDelayed(new Runnable() {

public void run() {

Loaddialog2.dismiss();

}

}, 500);

this.mPrompt.setVisibility(0);

this.mAddr1.setVisibility(0);

this.mAddr2.setVisibility(0);

this.mMyLinear.scrollTo(0, 0);

} else {

this.finalServer.suspend();

this.toolbar.setBackgroundColor(ResourcesCompat.getColor(getResources(), C0463R.color.colorRed, (Resources.Theme) null));

this.mBTimageViewir\_1.setVisibility(4);

this.mBTimageViewir\_2.setVisibility(4);

this.mBTimageViewir\_3.setVisibility(4);

this.mBTimageViewled.setVisibility(4);

this.mPrompt.setVisibility(4);

this.mAddr1.setVisibility(4);

this.mAddr2.setVisibility(4);

if (this.SCR\_CNT >= 1) {

this.mWebViewIR1.setVisibility(8);

}

if (this.SCR\_CNT >= 2) {

this.mWebViewIR2.setVisibility(8);

}

if (this.SCR\_CNT >= 3) {

this.mWebViewIR3.setVisibility(8);

}

if (this.SCR\_CNT >= 4) {

this.mWebViewIR4.setVisibility(8);

}

if (this.SCR\_CNT >= 5) {

this.mWebViewIR5.setVisibility(8);

}

findViewById(C0463R.C0465id.BTimageView11).setVisibility(8);

findViewById(C0463R.C0465id.BTimageView12).setVisibility(8);

findViewById(C0463R.C0465id.BTimageView13).setVisibility(8);

findViewById(C0463R.C0465id.BTimageView14).setVisibility(8);

findViewById(C0463R.C0465id.BTimageView15).setVisibility(8);

this.mMyLinear.scrollTo(0, 0);

}

}

public /\* synthetic \*/ void lambda$serverControl$6$MainActivity(DialogInterface dialog, int id) {

dialog.dismiss();

this.justStarted = false;

ActivityCompat.requestPermissions(this, new String[]{"android.permission.WRITE\_EXTERNAL\_STORAGE"}, 2203);

}

public void opw(View view) {

final WebView wv = new WebView(this);

String ss = ((Button) view).getText().toString();

final int k = Integer.parseInt(view.getTag().toString()) - 1;

wv.getSettings().setJavaScriptEnabled(true);

wv.setWebViewClient(new WebViewClient() {

public void onPageFinished(WebView view, String url) {

super.onPageFinished(view, url);

WebView webView = wv;

webView.loadUrl("http://" + MainActivity.this.SCR\_ARR[k] + "/liveonly");

}

});

wv.loadUrl("http://" + this.SCR\_ARR[k] + "/liveonly/admin:admin");

AlertDialog.Builder builder = new AlertDialog.Builder(this);

FrameLayout container = new FrameLayout(this);

FrameLayout.LayoutParams params = new FrameLayout.LayoutParams(-1, -1);

params.height = (int) TypedValue.applyDimension(1, ((float) getResources().getDisplayMetrics().heightPixels) \* 1.8f, getResources().getDisplayMetrics());

params.width = (int) TypedValue.applyDimension(1, ((float) getResources().getDisplayMetrics().widthPixels) \* 1.8f, getResources().getDisplayMetrics());

wv.setLayoutParams(new FrameLayout.LayoutParams(-1, -1));

wv.setLayoutParams(params);

wv.setInitialScale(50);

wv.getSettings().setUseWideViewPort(true);

container.addView(wv);

builder.setView((View) container);

builder.setTitle((CharSequence) "Live Cam " + ss);

builder.setNegativeButton((CharSequence) "Close", (DialogInterface.OnClickListener) new DialogInterface.OnClickListener() {

public void onClick(DialogInterface dialog, int id) {

dialog.dismiss();

}

});

AlertDialog dialog = builder.create();

WindowManager.LayoutParams layoutParams = new WindowManager.LayoutParams();

layoutParams.copyFrom(dialog.getWindow().getAttributes());

layoutParams.width = -1;

layoutParams.height = -1;

dialog.getWindow().setAttributes(layoutParams);

params.height = (int) TypedValue.applyDimension(1, ((float) getResources().getDisplayMetrics().heightPixels) \* 0.8f, getResources().getDisplayMetrics());

params.width = (int) TypedValue.applyDimension(1, ((float) getResources().getDisplayMetrics().widthPixels) \* 0.8f, getResources().getDisplayMetrics());

dialog.show();

}

private class TraceFtp extends DefaultFtplet {

private TraceFtp() {

}

public FtpletResult onConnect(FtpSession session) throws FtpException, IOException {

FtpletResult ftpletResult = FtpletResult.DISCONNECT;

MainActivity.this.client\_ip = session.getClientAddress().getAddress().getHostAddress();

Log.e("TAG\_CLIENT\_IP", MainActivity.this.client\_ip);

return FtpletResult.DEFAULT;

}

public FtpletResult onUploadEnd(FtpSession session, FtpRequest request) throws FtpException, IOException {

String path = session.getUser().getHomeDirectory();

String name = session.getUser().getName();

String filename = request.getArgument();

Log.e("FTPCOME", "REQ" + request.getCommand() + request.getArgument() + request.getRequestLine());

StringBuilder sb = new StringBuilder();

sb.append("SES");

sb.append(session.getUserArgument());

sb.append(session.toString());

Log.e("FTPCOME", sb.toString());

try {

MainActivity.this.givenUsingExecutorService\_whenSchedulingRepeatedTask\_thenCorrect(path, filename);

} catch (InterruptedException e) {

e.printStackTrace();

}

return super.onUploadEnd(session, request);

}

}

public void givenUsingExecutorService\_whenSchedulingRepeatedTask\_thenCorrect(final String p, final String f) throws InterruptedException {

new Handler(Looper.getMainLooper()).postDelayed(new Runnable() {

public void run() {

Log.e("RUN\_TIMER", "");

MainActivity.this.alarmcomes(p, f);

}

}, 1000);

}

public void BTimageViewled\_onClick(View v) {

String dd = "";

String don = "";

if (this.mBTimageViewLED\_flag == 0) {

this.mBTimageViewLED\_flag = 1;

} else {

this.mBTimageViewLED\_flag = 0;

}

if (this.mBTimageViewLED\_flag == 0) {

dd = "False";

this.mBTimageViewled.setImageResource(C0463R.C0464drawable.led\_btn);

}

if (this.mBTimageViewLED\_flag == 1) {

dd = "True";

this.mBTimageViewled.setImageResource(C0463R.C0464drawable.led\_btn\_on);

}

for (int k = 0; k <= this.SCR\_CNT - 1; k++) {

String ff = "http://" + this.SCR\_ARR[k] + "/res.php";

try {

don = sendPOST(ff, "action=set&resource=.system.vcam.torch&value=" + dd);

Log.e("LAMPON", ff + don);

} catch (Exception ex) {

Log.e("LAMPON", "hata" + ff + don + ex.toString());

}

}

}

public void BTimageViewir\_tag\_onClick(final View v) {

final ProgressDialog Loaddialog = ProgressDialog.show(this, "", "Please wait...", true);

Handler handler = new Handler(Looper.getMainLooper());

handler.postDelayed(new Runnable() {

public void run() {

int rr = Integer.parseInt(v.getTag().toString());

String dd = "";

if (rr == 1) {

dd = "IR";

}

if (rr == 2) {

dd = "FUSION";

}

if (rr == 3) {

dd = "VISUAL";

}

for (int k = 0; k <= MainActivity.this.SCR\_CNT - 1; k++) {

String ff = "http://" + MainActivity.this.SCR\_ARR[k] + "/login/dologin";

String don = "";

try {

don = MainActivity.sendPOST(ff, "user\_name=admin&user\_password=admin");

Log.e("LOGIN", ff + don);

} catch (Exception ex) {

Log.e("LOGIN", "hata" + ff + ex.toString());

}

try {

String ff2 = "http://" + MainActivity.this.SCR\_ARR[k] + "/home/setviewmode/" + dd;

don = MainActivity.sendGET(ff2);

Log.e("IRBUTTON", ff2 + don);

} catch (Exception e) {

Log.e("IRBUTTON", "hata" + don);

}

}

}

}, 100);

handler.postDelayed(new Runnable() {

public void run() {

Loaddialog.dismiss();

}

}, 500);

}

public void BTimageViewir\_onClick(View v) {

String dd = "";

String don = "";

if (this.mBTimageViewIR\_flag == 0) {

this.mBTimageViewIR\_flag = 1;

} else {

this.mBTimageViewIR\_flag = 0;

}

if (this.mBTimageViewIR\_flag == 0) {

if (this.mBTimageViewIRLAST == 0) {

this.mBTimageViewIRLAST = 1;

dd = "IR";

} else {

this.mBTimageViewIRLAST = 0;

dd = "FUSION";

}

this.mBTimageViewir.setImageResource(C0463R.C0464drawable.ir\_btn\_on);

}

if (this.mBTimageViewIR\_flag == 1) {

dd = "VISUAL";

this.mBTimageViewir.setImageResource(C0463R.C0464drawable.ir\_btn);

}

for (int k = 0; k <= this.SCR\_CNT - 1; k++) {

try {

String ff = "http://" + this.SCR\_ARR[k] + "/home/setviewmode/" + dd;

don = sendGET(ff);

Log.e("IRBUTTON", ff + don);

} catch (Exception e) {

Log.e("IRBUTTON", "hata" + don);

}

}

}

public void mBTimageView\_onClick(View v) {

Button button = (Button) findViewById(C0463R.C0465id.BTimageView);

this.mMyLinear.invalidate();

this.mMyLinear.requestLayout();

int i = this.BTTAG;

if (i == 0) {

Log.e("BUTTON", "click" + v);

this.mediaPlayer.pause();

this.mMyLinear.invalidate();

this.mMyLinear.requestLayout();

Log.e("BUTTON", "click end" + v);

this.BTTAG = 1;

} else if (i == 1) {

Log.e("BUTTON", "click beg" + v);

Log.e("BUTTON", "click end" + v);

this.BTTAG = 0;

}

this.mMyLinear.scrollTo(0, 0);

}

public static String sendPOST(String command, String data) throws IOException {

String str = command;

new StringBuilder("");

Log.e("sendget0\_command", "" + str);

HttpURLConnection con = (HttpURLConnection) new URL(str).openConnection();

byte[] postData = data.getBytes(StandardCharsets.UTF\_8);

int postDataLength = postData.length;

con.setConnectTimeout(1500);

con.setReadTimeout(1500);

con.setDoOutput(true);

con.setInstanceFollowRedirects(false);

con.setRequestMethod("POST");

con.setRequestProperty("Content-Type", "application/x-www-form-urlencoded");

con.setRequestProperty("charset", "utf-8");

con.setRequestProperty("Content-Length", Integer.toString(postDataLength));

con.setUseCaches(false);

DataOutputStream wr = new DataOutputStream(con.getOutputStream());

try {

wr.write(postData);

wr.close();

int responseCode = con.getResponseCode();

Log.e("sendget", "" + responseCode);

if (responseCode == 200) {

BufferedReader in = new BufferedReader(new InputStreamReader(con.getInputStream()));

StringBuffer response = new StringBuffer();

while (true) {

String readLine = in.readLine();

String inputLine = readLine;

if (readLine == null) {

break;

}

response.append(inputLine);

}

StringBuilder sb = new StringBuilder();

List<String> cookies = (List) con.getHeaderFields().get("Set-Cookie");

if (cookies != null) {

for (String cookie : cookies) {

if (sb.length() > 0) {

sb.append("; ");

}

sb.append(cookie.split(";")[0]);

String str2 = command;

}

}

String cookieHeader = sb.toString();

in.close();

Log.e("coookie\_post", "" + cookieHeader);

System.out.println(response.toString());

return response.toString();

}

System.out.println("GET request not worked");

System.out.println("-----------------------");

return "".toString();

} catch (Throwable th) {

Throwable th2 = th;

try {

wr.close();

} catch (Throwable th3) {

}

throw th2;

}

}

public static String sendGET(String command) throws IOException {

new StringBuilder("");

Log.e("sendget0\_command", "" + command);

HttpURLConnection con = (HttpURLConnection) new URL(command).openConnection();

con.setConnectTimeout(1500);

con.setReadTimeout(1500);

con.setRequestMethod(HttpProxyConstants.GET);

con.setRequestProperty("User-Agent", USER\_AGENT);

int responseCode = con.getResponseCode();

Log.e("sendget", "" + responseCode);

if (responseCode == 200) {

BufferedReader in = new BufferedReader(new InputStreamReader(con.getInputStream()));

StringBuffer response = new StringBuffer();

while (true) {

String readLine = in.readLine();

String inputLine = readLine;

if (readLine != null) {

response.append(inputLine);

} else {

in.close();

System.out.println(response.toString());

return response.toString();

}

}

} else {

System.out.println("GET request not worked");

System.out.println("-----------------------");

return "".toString();

}

}

/\* access modifiers changed from: package-private \*/

public void alarmcomes(String path, String file) {

((Button) findViewById(C0463R.C0465id.BTimageView)).setVisibility(0);

LinearLayout mMyLinear2 = (LinearLayout) findViewById(C0463R.C0465id.MyLinear);

mMyLinear2.invalidate();

mMyLinear2.requestLayout();

((ImageView) findViewById(C0463R.C0465id.imageView\_last)).setImageURI(Uri.fromFile(new File(path, file)));

Log.e("TAG", "aaaa\_pass" + path + file);

Log.e("TAG\_CLIENT\_IP", this.client\_ip);

if (!this.mediaPlayer.isPlaying()) {

this.mediaPlayer.start();

this.mediaPlayer.setLooping(true);

}

}

public void showExpire() {

if (this.progress == null) {

this.progress = ProgressDialog.show(this, "This version Expired", "Please load new Version", true, true);

}

}

public void showLoadingDialog() {

if (this.progress == null) {

this.progress = ProgressDialog.show(this, "Searching...", "Please wait while loading client photos...", true, true);

}

}

public void dismissLoadingDialog() {

ProgressDialog progressDialog = this.progress;

if (progressDialog != null && progressDialog.isShowing()) {

this.progress.dismiss();

}

}

}