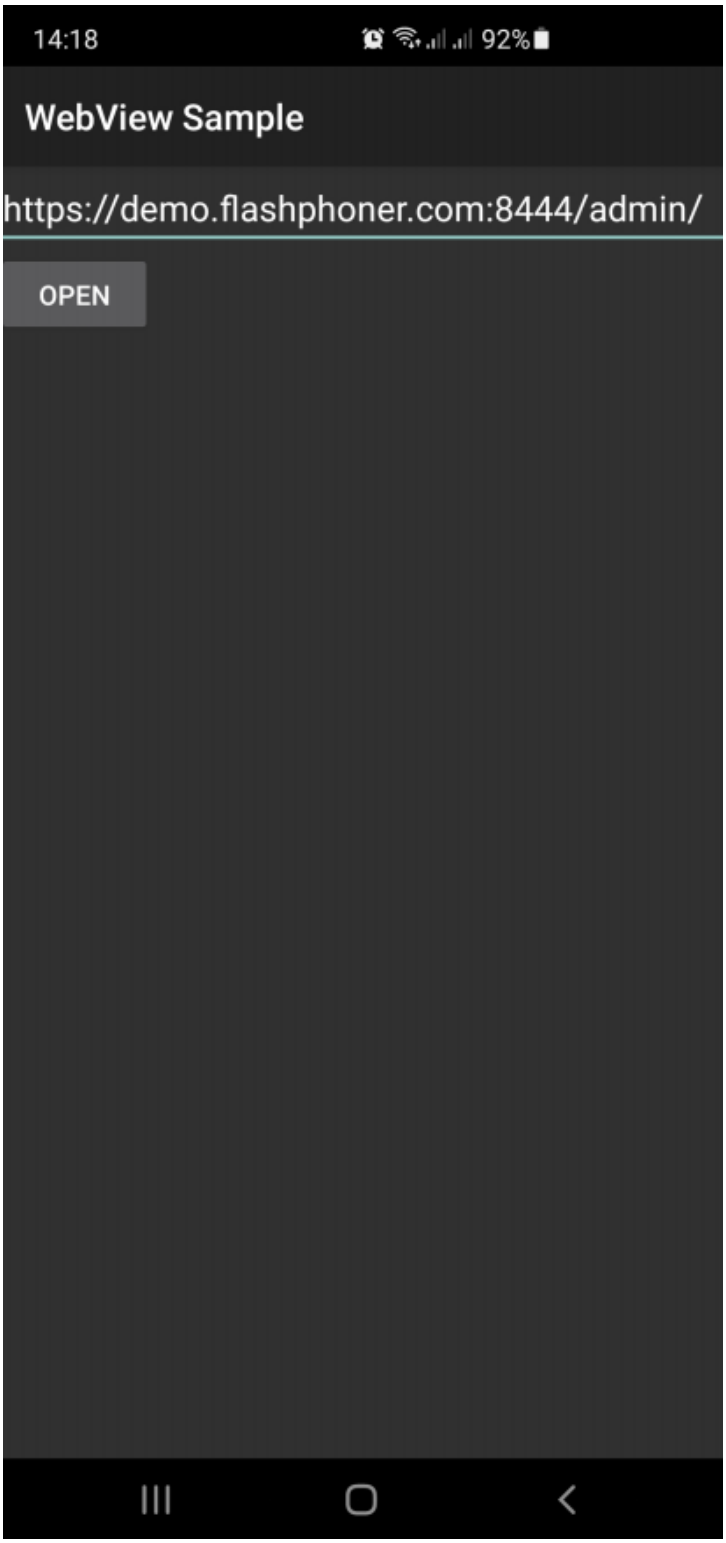
Android Webview

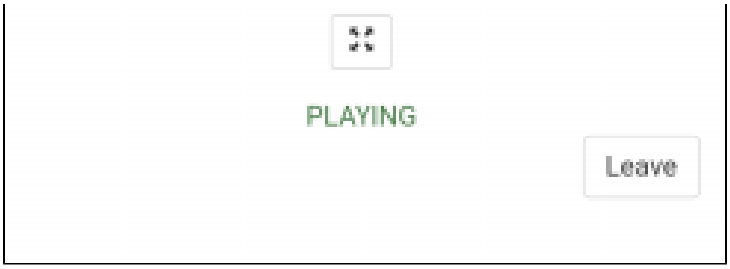
Android WebView application example

This example can be used to open anyWeb SDK example page like in a bowser.To open the page, the URL should be set



MCU Client example opened in Android Webview in audio only conference mode looks as follows

14:22
91%
MCU Client
Before use: please set the server parameters as described here
Conference
WCS URL
wss://demo.flashphoner.com:8443
Login
user1
Room
room1
Volume
Audio
Full Screen



Analyzing the code

To analyze the code takeMainActivity.javaandWebViewActivity.javaclasses of the webview-example application, which is available in build1.1.0.26.

1. Launch WebViewActivity with URL entered

code

```
        button = (Button) findViewById(R.id.btnUrl);
        button.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                EditText et = (EditText) findViewById(R.id.textUrl);
                String url = et.getText().toString();
                if (url.isEmpty()) {
                    showDialog();
                } else {
                    Intent intent = new Intent(context, WebViewActivity.class);
                    intent.putExtra("url", url);
                    startActivity(intent);
                }
            }
        });
```

2. WebView settings configuration

code

```
        WebSettings settings = webView.getSettings();

        // Enable Javascript
        settings.setJavaScriptEnabled(true);

        // Use WideViewport and Zoom out if there is no viewport defined
        settings.setUseWideViewPort(true);
        settings.setLoadWithOverviewMode(true);

        // Enable pinch to zoom without the zoom buttons
        settings.setBuiltInZoomControls(true);

        // Allow use of Local Storage
        settings.setDomStorageEnabled(true);

        if(Build.VERSION.SDK_INT > Build.VERSION_CODES.HONEYCOMB) {
            // Hide the zoom controls for HONEYCOMB+
            settings.setDisplayZoomControls(false);
        }

        // Enable remote debugging via chrome://inspect
        if(Build.VERSION.SDK_INT >= Build.VERSION_CODES.KITKAT) {
            WebView.setWebContentsDebuggingEnabled(true);
        }
```

3. SSL certificates error handler configuration

code

```
        webView.setWebViewClient(new WebViewClient() {
            @Override
            public void onReceivedSslError(WebView view, final SslErrorHandler handler, SslError error) {

                final AlertDialog.Builder builder = new AlertDialog.Builder(view.getContext());
                String message = "SSL Certificate error.";
                switch (error.getPrimaryError()) {
                    case SslError.SSL_UNTRUSTED:
                        message = "The certificate authority is not trusted";
                        break;
                    case SslError.SSL_EXPIRED:
                        message = "The certificate has expired";
                        break;
                    case SslError.SSL_NOTYETVALID:
                        message = "The certificate is not yet valid.";
                        break;
                    case SslError.SSL_IDMISMATCH:
                        message = "The cerificate ID is mismatch";
                        break;
                    case SslError.SSL_DATE_INVALID:
                        message = "The certificate date is invalid";
                        break;
                    case SslError.SSL_INVALID:
                        message = "The certificate is invalid";
                        break;
                }
                builder.setTitle("SSL Cerificate Error");
                builder.setMessage(message);
                builder.setPositiveButton("Continue", new DialogInterface.OnClickListener() {
                    @Override
                    public void onClick(DialogInterface dialogInterface, int i) {
                        handler.proceed();
                    }
                });
                builder.setNegativeButton("Cancel", new DialogInterface.OnClickListener() {
                    @Override
                    public void onClick(DialogInterface dialogInterface, int i) {
                        handler.cancel();
                    }
                });
                Log.d(TAG, "onReceivedSslError " + message);
                final AlertDialog dialog = builder.create();
                dialog.show();
            }
        });
    }
```

4. Camera and microphone access requesting handler configuration

code

```
        webView.setWebChromeClient(new WebChromeClient() {
            @Override
            public void onPermissionRequest(final PermissionRequest request) {
                Log.d(TAG, "Request permissions: ");
                for (String res : request.getResources()) {
                    Log.d(TAG, res);
                }
                WebViewActivity.this.runOnUiThread(new Runnable() {
                    @TargetApi(Build.VERSION_CODES.LOLLIPOP)
                    @Override
                    public void run() {
                        request.grant(request.getResources());
                    }
                });
            }

            @Override
            public void onPermissionRequestCanceled(PermissionRequest request) {
                Log.d(TAG, "onPermissionRequestCanceled");
            }

        });
```

5. Requesting camera and microphone access on activity start

code

```
        if (Build.VERSION.SDK_INT >= Build.VERSION_CODES.M) {
            int hasCameraPermission = checkSelfPermission(Manifest.permission.CAMERA);
            int hasRecordPermission = checkSelfPermission(Manifest.permission.RECORD_AUDIO);
            List<String> permissions = new ArrayList<>();
            if (hasCameraPermission != PackageManager.PERMISSION_GRANTED) {
                permissions.add(Manifest.permission.CAMERA);
            }
            if (hasRecordPermission != PackageManager.PERMISSION_GRANTED) {
                permissions.add(Manifest.permission.RECORD_AUDIO);
            }
            if (!permissions.isEmpty()) {
                requestPermissions(permissions.toArray(new String[permissions.size()]), 111);
            }
        }
```

6. URL opening

code

```
        webView.loadUrl(url);
```