

# Controlarea aplicațiilor ce funcționează pe sisteme Windows prin intermediul dispozitivelor Android

Cristina Șerban, Irina Grosu, Alexandra Sirițeanu, Alexandru Averescu, Adrian Iftene

Universitatea “Alexandru Ioan Cuza” din Iași, Facultatea de Informatică

{cristina.serban, irina.grosu, alexandra.siriteanu, alexandru.averescu, adiftene}@info.uaic.ro

## REZUMAT

Articolul de față prezintă o aplicație creată de autorii articolului pentru platforma Android, care permite unui utilizator să controleze de la distanță aplicațiile ce funcționează pe un calculator pe care rulează sistemul de operare Microsoft Windows. Aplicația este compusă din două module importante: *clientul* (aplicația ce funcționează pe telefonul utilizatorului) și *serverul* (aplicația ce funcționează pe calculator), la care se adaugă protocoalele de comunicare dintre cele două module (bazate pe Bluetooth și pe Wi-Fi). În cele ce urmează, vom face o prezentare detaliată a arhitecturii sistemului, urmată de o evaluare a acestei soluții, bazată pe câteva studii de caz și pe câteva teste relevante.

## Cuvinte cheie

Android, servicii Windows, control la distanță.

## MOTIVAȚIE

În ultimul timp am asistat la o explozie a diversității dispozitivelor electronice ce ne permit să ne conectăm la Internet pentru a citi știrile, pentru a ne citi e-mailul sau pentru a socializa prin intermediul rețelelor sociale sau pentru a asculta muzică sau pentru a vizualiza prezentări de tip PDF sau Power Point (laptop-uri, notebook-uri, palmtop, etc.). La toate aceste dispozitive în cazul unor prezentări la conferințe sau în cazul în care ascultăm muzică am dorit să oferim posibilitatea de a controla astfel de aplicații de la distanță.

În prezent, asistăm la o apariția a multor instrumente de acest tip, vezi de exemplu Vectir [1], o aplicație ce rulează pe dispozitive ce suportă Java, poate fi folosită gratuit pentru 30 de zile și permite controlul prin Bluetooth și Wi-Fi a aplicațiilor precum Winamp, Windows Media Player, iTunes și PowerPoint. PPT Remote [2] este o aplicație dezvoltată pentru telefoanele mobile ce utilizează sistemul de operare Android și pot fi conectate la calculator prin Wi-Fi sau Bluetooth, oferind diverse moduri de a controla prezentări de tip Power Point. O altă aplicație, Remote Presenter for Android [3] utilizează capacitățile tehnologiei Bluetooth pentru a controla de la distanță prezentări Microsoft Power Point, Keynote, OpenOffice și fișiere PDF, fiind compatibilă cu sistemele de operare Windows, Mac și Linux. Un exemplu de aplicație de tip iOS este Tango Remote Control Media Player HD [4], disponibilă în 16 limbi și menită să controleze prin Wi-Fi sau Bluetooth aplicația iTunes pentru Mac.

Aplicația pe care o vom prezenta în continuare, permite controlarea prezentărilor PowerPoint și a aplicației

Windows Media Player atât prin Bluetooth cât și prin Wi-Fi. Prin intermediul touchscreen-ului telefonului dumneavoastră puteți porni o prezentare Power Point sau puteți trece la slide-urile următoare sau la cele anterioare. Similar, în cazul aplicației Media Player puteți asculta lista favorită de melodii, puteți schimba melodiile care sunt redade sau puteți modifica nivelul volumului. Pentru toate acestea nu mai este nevoie să fiți în fața calculatorului, ci o puteți face mult mai ușor de oriunde v-ați afla, folosind ecranul touchscreen al telefonului dumneavoastră.

În cele ce urmează vom prezenta pe scurt soluția pe care am construit-o, oprindu-ne asupra elementelor sale principale, componenta client și componenta server. În partea de final facem o evaluare a acestui sistem, prezentând cele mai importante avantaje și dezavantaje.

## PREZENTAREA SOLUȚIEI

### Arhitectura

După cum putem observa din Figura 1, aplicația este compusă din două părți principale: *componenta client* (care e reprezentată de aplicația Android ce funcționează pe telefon) și *componenta server* (care e reprezentată de aplicația Desktop ce rulează pe sistemul de operare Windows). *Comunicarea* dintre acestea poate fi realizată prin Bluetooth și Wi-Fi.

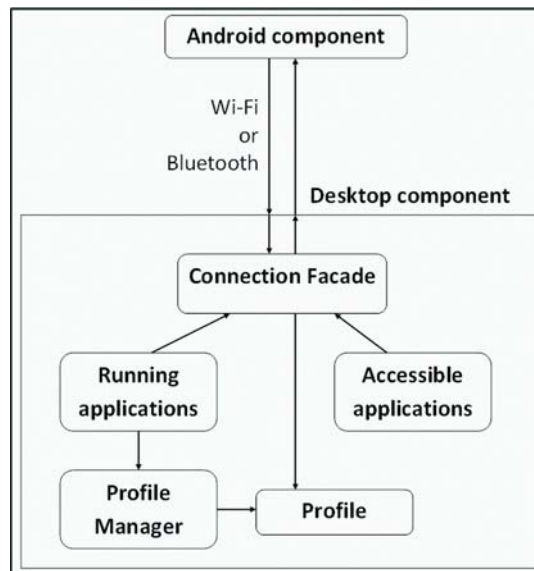


Figura 1: Arhitectura sistemului

## Componenta Client

Am dezvoltat o aplicație Android care se poate conecta la orice PC pe care rulează aplicația server. Clientul este construit urmând indicațiile și principiile de design pentru Android UI [7], asigurând o experiență plăcută și o interacțiune ergonomică între utilizator și dispozitivul mobil.

Aplicația Android constă din două ecrane, organizate în tab-uri pentru a oferi simplitate (vezi Figura 2 de mai jos).

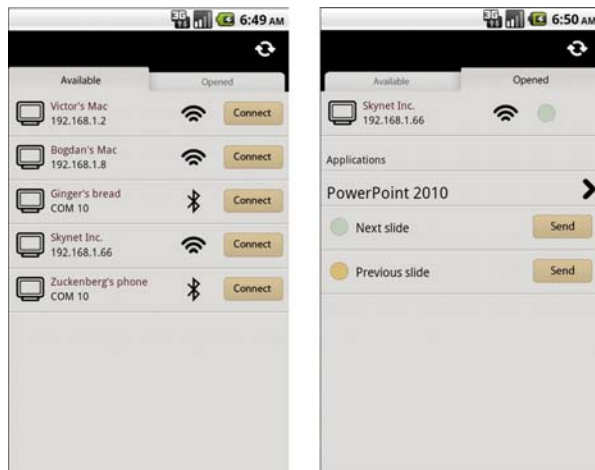


Figura 2: Ecranele aplicației Android

Primul ecran afișează o listă a dispozitivelor la care se poate conecta clientul, lucru care se poate realiza apăsând butonul de conectare (Figura 1 în stânga). Mai mult, sunt oferite reprezentări grafice ale tipului de conexiune ce se poate realiza (Bluetooth sau Wi-Fi), alături de adresa IP și numele calculatorului. Odată conectat, al doilea ecran devine disponibil și oferă informații asupra aplicațiilor care pot fi controlate (Figura 2 în dreapta). Este afișată o listă conținând toate comenzile ce pot fi trimise la server prin apăsarea butoanelor corespunzătoare. Trecerea de la o aplicație la alta se poate realiza prin apăsarea unei săgeți din dreptul numelui aplicației curente.

În partea de sus, este disponibil un buton de refresh pentru ca, atunci când utilizatorul dorește, să se poată actualiza lista de dispozitive cu care se poate stabili o conexiune.

## Componenta Server

Componenta desktop conține trei module principale. Modulul „Accessible Applications”, care conține o listă de aplicații ce pot fi accesate de programul nostru. Informațiile despre aplicațiile suportate sunt preluate dintr-un fișier de configurare *.xml*. Modulul „Running Applications” monitorizează procesele ce rulează pe un calculator. Rolul acestuia este să notifice telefonul conectat la program de fiecare dată când o aplicație este deschisă sau închisă. Modulul „Profile Manager” creează profile specifice pentru o aplicație, prin intermediul cărora poate realiza diferite acțiuni (e.g. run, exit, close presentation, view show, exit show, next slide, și previous slide pentru aplicația PowerPoint).

Acest modul a fost scris în C# și este programat să pornească odată cu calculatorul. Aplicația rulează în

background, putând fi accesată de către utilizator din *system tray* și odată pornită așteaptă ca un telefon să se conecteze prin Wi-Fi sau Bluetooth. La un moment dat, doar un singur dispozitiv mobil poate fi conectat la această aplicație.

Componenta server primește comenzi de la componenta Android și astfel putem controla aplicații precum Microsoft Power Point și Windows Media Player. Lista aplicațiilor accesibile și acțiunile suportate de către fiecare dintre acestea este stabilită la runtime, prin citirea informațiilor aferente din fișierul de configurare *.xml*. Solicitarea execuției unei acțiuni înseamnă recepționarea unui șir de caractere codificat de la client. Validitatea este verificată prin utilizarea de expresii regulate. În cazul în care comanda nu se potrivește cu formatul existent, este afișat un mesaj de eroare. Dacă mesajul trece de verificare, dar nu este găsit în lista de comenzi disponibile, cererea este ignorată și este afișat un mesaj de eroare specific. Dacă niciuna dintre aceste situații nu are loc, atunci comanda este corectă și acțiunea corespunzătoare este executată.

Modulul „Running Applications” monitorizează procesele din sistem și întreprinde acțiuni de fiecare dată când are loc o modificare relevantă (e.g. un proces pornește sau se termină). El actualizează lista cu aplicațiile active și notifică componenta Android.

Modulul „Profile Manager” ne permite să creăm profile, fiecare specializat pentru un anumit tip de aplicație. Putem reprezenta orice tip de aplicație pe care sistemul o poate suporta. Asignându-i un anumit profil, creăm un „bridge” și, astfel, orice comandă primită de aplicație ajunge în profil, iar acesta știe exact ce acțiune să întreprindă.

Microsoft PowerPoint este controlat obținând obiectul aferent aplicației active din memorie, sau prin crearea acestui obiect. Pentru aceasta, folosim biblioteca Microsoft.Office.Interop [5], oferită de platforma .NET. Aplicația noastră poate identifica și acțiunile inițiate de un utilizator direct pe calculator (e.g. deschiderea manuală a unei prezentări). Windows Media Player este controlat trimițând comenzi de tip scurtătură ferestrei active.

Când dispozitivul Android este deconectat, programul desktop începe să aștepte să primească noi cereri de conectare.

## Comunicarea

Modulul pentru comunicarea server-client este o componentă foarte importantă și oferă posibilitatea trimiterii și primirii de mesaje între server și client.

Comunicarea este reprezentată prin trimiterea de șiruri de caractere cu un format prestabilit între client (componenta Android) și server (componenta desktop). Comunicarea poate fi realizată în două moduri, în funcție de preferințele utilizatorului: folosind Bluetooth sau folosind Wi-Fi.

### Comunicarea folosind Bluetooth

Prima modalitate de comunicare poate fi realizată prin intermediul conexiunii bluetooth. Modulul „Bluetooth” oferă o alternativă pentru trimiterea mesajelor și comenzilor pentru aplicația pe care am construit-o. Este foarte important să avem o altă metodă de comunicare acolo unde nu există acces la Internet sau când utilizatorul nu dispune de

conexiune Wi-Fi. Aplicația oferă aceleași facilități indiferent de metoda aleasă pentru comunicație.

- *Identificarea dispozitivelor și stabilirea conexiunii*

Pentru a crea o conexiune între cele două dispozitive, serverul trebuie să verifice dacă telefonul Android are un adaptor Bluetooth încorporat. Dacă adaptorul este pornit, utilizatorul poate căuta componente desktop la care să se conecteze. Pentru a realiza acest lucru, trebuie să introducă codul de securitate dat de PC într-un timp limită. Dacă nu se stabilește o conexiune în acest timp, codul este resetat și se poate încerca din nou.

- *Trimiterea de mesaje și comenzi*

Dacă s-a stabilit cu succes o conexiune, funcțiile de *read* și *write* pot fi folosite pentru a trimite mesaje. Pentru a evita blocarea interfeței grafice, funcția de citire rulează într-un thread separat, fiind întreruptă atunci când conexiunea prin socket aruncă o excepție. Acest lucru face ca serverul să termine bucla de ascultare, oferind posibilitatea de conectare și altor dispozitive mobile. Un singur client poate fi conectat la un moment dat la server, făcând aplicația mai sigură.

#### *Comunicarea folosind Wi-Fi*

A doua modalitate de comunicare se realizează prin intermediul conexiunii wireless. Modulul „Wireless Connectivity” oferă două facilități celorlalte module:

- 1) *Clientul* va avea posibilitatea de a identifica serverele din apropiere care rulează aplicația și care îi oferă posibilitatea de a se conecta la ele.
- 2) *Clientul și serverul* trebuie să poată trimite și primi mesaje după ce conexiunea dintre ei a fost stabilită.

Pentru a putea oferi prima facilitate, serverul trebuie să trimită un mesaj care conține adresa IP a acestuia. Orice client care se află în raza de acțiune a serverului va aștepta mesajul și va reține o listă cu toate serverele disponibile. Este stabilită o perioadă de așteptare, iar după ce aceasta s-a terminat, dacă serverul nu mai este identificat, nu va mai fi vizibil nici în lista oferită clientului.

A doua proprietate este implementată folosind o clasă „wrapper” pentru clasele Java: Socket și ServerSocket. Aceste clase vor oferi funcționalitățile necesare pentru a trimite și primi date, cât și pentru adresa IP și starea conexiunii. Acest lucru permite serverului și clientului să nu interacționeze direct cu conceptele de nivel jos.

Chiar dacă serverul este implementat în C#, modulul conexiunii wireless este implementat în Java. Folosind ikvm [6], am reușit să transformăm fișierul .jar într-un fișier valid .dll care poate fi folosit de către server pentru a comunica cu clientul prin trimiterea și primirea de mesaje.

#### **EVALUAREA SOLUȚIEI**

Spre deosebire de celelalte aplicații similare disponibile pe Internet, ne-am decis să ne concentrăm pe a realiza un produs ușor de utilizat. Deși o întreagă gamă de instrucțiuni sunt disponibile, acestea pot fi personalizate prin intermediul fișierului de configurare .xml, clientul Android având acces doar la cele existente în fișier.

Alternarea aplicațiilor poate fi realizată prin simpla apăsare a unui buton de pe ecran. Acest lucru este foarte util dacă se dorește combinarea unor fișiere video cu prezentarea existentă, tranziția dintre ele fiind atât rapidă, cât și ușor de manevrat, ceea ce oferă o facilitate suplimentară utilizatorului.

#### **Studii de caz și teste**

În scopul evaluării soluției noastre și al monitorizării comportamentului ei în diverse situații, am realizat o serie de teste și studii de caz.

##### *Utilizarea aplicației timp îndelungat*

Pe parcursul mai multor teste, am utilizat programul timp de aproximativ 40 de minute, susținând o prezentare Power Point, și nu am întâmpinat probleme majore. Aplicația s-a comportat cum era de așteptat, permițându-ne să ne deplasăm înainte și înapoi prin documentul Power Point. Legătura dintre client și server, atât prin intermediul conexiunii Bluetooth, cât și a conexiunii Wi-Fi, a putut fi deschisă la orice moment și nu au existat situații neprevăzute, cum ar fi întreruperea bruscă sau probleme de conectare, atât pe partea clientului, cât și a serverului.

##### *Variația distanței*

În cazul acestor teste am modificat în timpul prezentărilor distanța dintre calculator și telefonul mobil pe care aveam aplicația Android. În timp ce distanța dintre telefon și calculator nu este o problemă atunci când este implicată conexiunea Wi-Fi, pentru Bluetooth aceasta reprezintă un element important, de care trebuie să ținem cont. Pentru a vedea care este aria de acoperire, am testat pe o distanță cuprinsă între 0 și 25 de metri. Am ajuns la concluzia că aplicația rulează fără dificultăți pe o distanță de maxim 20 de metri, peste acest prag existând posibilitatea deconectării (cam în 60% din cazurile testate conexiunea s-a întrerupt).

##### *Obstacole*

De această dată, am luat în considerare situațiile când între client și server există diverse obiecte (am considerat chiar și situația când cele două se află în camere diferite, despărțite printr-un perete) și am verificat cum influențează acestea comunicarea dintre ele. Și în cazul acestor scenarii, legătura prin Wi-Fi se realizează cu succes. Atunci când conexiunea se realizează prin Bluetooth, am observat următoarele: atât timp cât distanța nu depășește 20 de metri (care am dedus anterior că ar fi pragul de acceptare), programul funcționează fără probleme indiferent de micile obstacole care se interpun. Acest lucru este valabil și atunci când se încearcă stabilirea unei conexiuni din două camere diferite despărțite de un perete. Însă, la interpunerea a doi sau mai mulți pereți, conexiunea nu mai poate fi stabilită.

##### *Testele de stres*

Cu scopul de a detecta cât de multe instrucțiuni pot fi executate într-un timp redus, am realizat o testare manuală, cu ajutorul unui client ce apăsa foarte rapid butoanele din aplicația Android. Totodată, am creat un fișier jurnal cu fiecare instrucțiune executată de server împreună cu momentul când a fost apelată și timpul necesar executării ei. În urma analizării fișierului jurnal am observat că fiecare apăsare detectată de ecran a fost trimisă către server și fiecare comandă corespunzătoare acesteia a fost îndeplinită

cu succes. Într-o secundă am efectuat 7-8 instrucțiuni, un număr mai mare neputând fi posibil, întrucât testarea a fost realizată manual. Cu toate acestea, nu credem că acest tip de situație ar reprezenta o problemă pentru server, singura piedică putând fi constituită de ecranul telefonului în sine, în cazul în care nu ar detecta fiecare atingere. Acest lucru are loc atunci când frecvența mișcărilor este mai mare de 10 instrucțiuni pe secundă. Întrucât niciun utilizator nu ar fi capabil să execute un flux de comenzi atât de rapid, considerăm că aplicația funcționează corect în acest tip de situații.

#### *Versiunile software*

Programul a fost testat pe versiunea 12 a aplicației Windows Media Player. În ceea ce privește Microsoft Power Point, aplicația a fost folosită pe versiunile 2003, 2007 și 2010 ale pachetului Microsoft Office. Trebuie să precizăm faptul că pe durata acestor teste nu am întâmpinat nicio problemă, nefiind necesare ajustări pentru niciuna dintre aceste versiuni.

#### **Avantaje**

- Soluție utilă pentru a accesa computerul de la distanță și pentru a executa comenzi pe acesta prin intermediul telefonului mobil;
- Rapidă și ușoară de folosit, putând înlocui cu succes o telecomandă clasică pentru controlul la distanță a prezentărilor;
- Poate fi extinsă pentru a controla și alte aplicații ce funcționează pe sistemul de operare Windows, în acest caz având loc o creștere a versatilității și a cazurilor de utilizare;
- Este o soluție de încredere în ceea ce privește stabilirea unei conexiuni și recepționarea de comenzi.

#### **Dezavantaje**

- Securitatea nu constituie un punct forte, neexistând o componentă ce verifică identitatea utilizatorului;
- Serverul poate rula doar pe calculatoarele pe care rulează Microsoft Windows, alte sisteme de operare nefiind suportate;
- Un alt dezavantaj este datorat ariei limitate de programe pe care aplicația le poate accesa.

#### **CONCLUZII ȘI DIRECȚII VIITOARE DE DEZVOLTARE**

Considerăm că am atins țelurile pe care ni le-am propus la începutul acestui proiect. Am creat o aplicație robustă, eficientă și ușor de utilizat care poate fi folosită pentru a controla de la distanță programele Windows Media Player și Microsoft Power Point.

În ceea ce privește o dezvoltare ulterioară, dorim ca modulul pentru server să-l facem să fie portabil pe orice platformă. Până în prezent, aplicațiile pe care le-am avut în vedere sunt specifice sistemului de operare Microsoft Windows, astfel încât o direcție importantă pentru

dezvoltarea produsului ar fi să considerăm aplicații care sunt independente de sistemul de operare (e.g. Open Office). Acest lucru ar face să crească numărul de situații în care aplicația noastră ar putea fi folosită.

Alte direcții ce ar putea fi exploatare în viitor ar putea fi:

- Adăugarea de elemente care să ne asigure securitatea soluției (e.g. autentificarea și verificarea identității), dând aplicației o notă de siguranță atunci când este vorba de prezentări oficiale și nu se dorește ca oricine să poată avea acces la aplicația desktop.
- Posibilitatea de a conecta simultan mai multe telefoane Android la server. Acest lucru ar putea fi posibil, dacă modul de gestionare al cererilor ar putea fi ușor ajustat, pentru a nu exista conflicte între ele.
- Studierea posibilității de a avea acces la toate aplicațiile deschise și oferirea posibilității utilizatorului de a controla aplicația dorită. Această cerere vine în contextul în care, chiar dacă serverul poate gestiona mai mult de o prezentare Power Point în același timp, identificarea având loc prin intermediul unui cod hash obținut prin aplicarea algoritmului SHA1 [8] pe locația fișierului, această funcționalitate nu este implementată și la nivel de client. Momentan, comanda pe care un client o trimite către aplicația desktop este direcționată către prezentarea Power Point activă.

#### **MULȚUMIRI**

Mulțumim colegilor de la Facultatea de Informatică care ne-au ajutat la construirea anumitor componente ale sistemului.

Cercetarea prezentată în această lucrare a fost finanțată de către Programul Operațional Sectorial Dezvoltarea Resurselor Umane prin proiectul „Dezvoltarea capacității de inovare și creșterea impactului cercetării prin programe post-doctorale POSDRU/89/1.5/S/49944”.

#### **REFERINȚE**

1. Vectir: <http://www.vectir.com/>
2. PPT Remote: <http://www.pptremotecontrol.com/>
3. Remote Presenter for Android: <http://www.free-power-point-templates.com/articles/remote-presenter-for-android/>
4. Tango Remote Control Media Player HD: [http://www.appsmenow.com/app\\_page/16978-Tango\\_Remote\\_Control\\_Media\\_Player\\_HD](http://www.appsmenow.com/app_page/16978-Tango_Remote_Control_Media_Player_HD)
5. Microsoft.Office.Interop.PowerPoint Namespace <http://msdn.microsoft.com/en-us/library/microsoft.office.interop.powerpoint.aspx>
6. IKVM.NET <http://www.ikvm.net/>
7. Android User Interface Guidelines [http://developer.android.com/guide/practices/ui\\_guidelines/index.html](http://developer.android.com/guide/practices/ui_guidelines/index.html)
8. Secure hash Standard. National Bureau of Standards, FIPS Publications 180, 1993