

RANCANG BANGUN APLIKASI PENGINTAI AKTIFITAS KOMPUTER MENGGUNAKAN LAYANAN CLOUD TO DEVICE MESSAGING (C2DM) PADA SMARTPHONE ANDROID

Azwar Anas¹⁾, Anjik Sukmaaji²⁾, Teguh Sutanto³⁾

¹⁾ Jurusan S1 Sistem Informasi STIKOM Surabaya

²⁾ Program studi Sistem Informasi STIKOM Surabaya, ³⁾ Program studi Sistem Informasi STIKOM Surabaya
e-mail: ¹⁾ azwar.nazz@gmail.com, ²⁾ anjik@stikom.edu, ³⁾ teguh@stikom.edu

Abstract: Based on The National Center For Education Statistic (NCES) survey, the number of people who own computers, spesifically kids and juvenile around five to seventeen years old, are moving up. There are two types of computers usage. First, Computers use as a facility to learn and study about homework tasks. Also, they are meant to be an entertainment media and information on the internet. Every activity use of computers, which is related to kids, is necessarily need to be controlled. Then, the system are made to actuate the activity use of computers by kids, which runs in background, and record the log history for their parents' importance. This goals are made the parents to actuate every activity use of computers by kids wherever they are without any bother to accompony. The result of this final project is log histories which classified based on date, username, and time sent to C2DM services on the Android smartphone, because it is able to connect to the internet simultaneously and synchronize the data in real time. To sum up, by the existance of the system, the parents' role to actuate the whole activity that related to their kids are not overlooked.

Keyword: Android, C2DM, Monitors Computer Activity, Keylogger

The National Center for Education Statistic (NCES) adalah entitas federal utama untuk mengumpulkan, menganalisa dan melaporkan data yang terkait dengan pendidikan di Amerika dan negar-negara lainnya. Untuk memenuhi mandate kongres dalam mengumpulkan, menyusun, menganalisa dan melaporkan penuh dan lengkap pada kondisi di Amerika. Melakukan dan mempublikasikan analisis laporan khusus serta membantu negara dan lembaga setempat dalam meningkatkan sistem statistik serta ulasan dan melaporkan kegiatan pendidikan di luar negeri.

Menurut hasil survei dari National Center Education Statistic (Debel, 2003), rumah dan sekolah adalah tempat utama anak-anak dan remaja menggunakan komputer, 81 persen dari anak usia (5-17) tahun menggunakan komputer disekolah dan 65 persen menggunakan komputer dirumah secara umum penggunaan komputer disekolah lebih banyak daripada dirumah. Segmentasi penggunaan internet pada remaja di rentang usia ini sebanyak 42 menggunakannya untuk menyelesaikan tugas sekolah, sementara 38 persen menggunakan sarana email atau chatting termasuk didalamnya adalah bermain game online, dan 22 persen selanjutnya menggunakan internet untuk browsing dan mencari informasi.

Mengingat dari waktu ke waktu, kepemilikan komputer semakin meningkat tiap tahun, sedikit banyak akan mempengaruhi jumlah pengguna yang berinteraksi dengannya. Berbagai-macam konten di internet

misalnya, tidak semua konten berisi hal-hal berbau pendidikan maupun pengetahuan yang bisa digunakan sebagai wawasan belajar, namun banyak juga konten - konten yang berisi informasi negatif yang tidak selayaknya diakses oleh seorang anak dan remaja. Oleh karena itu, akibat dari meningkatnya peran komputer rumah dan informasi yang beragam dalam kehidupan anak-anak maka diperlukan sebuah perhatian khusus yang tidak boleh lepas dari orang tua. Maka dibuatkan sebuah sistem yang mampu memantau aktifitas anak saat mengakses komputer dirumah atau dimanapun berada menggunakan aplikasi pada perangkat mobile, aktifitas yang direkam adalah seluruh kegiatan penggunaan komputer berupa log / history dari input keyboard dan screencapture. Dengan aplikasi mobile orang tua yang sibuk dengan pekerjaannya maupun sedang berpergian tetap masih dapat memantau aktifitas penggunaan komputer anak tanpa harus selalu mendampingi.

METODE

Keylogger

Keylogger adalah program untuk memonitor (memantau) segala aktifitas yang dilakukan oleh pengguna komputer, jenis keylogger yang digunakan pada sistem ini adalah keylogger software berupa aplikasi yang nantinya akan diinstal pada komputer yang mampu merekam segala inputan dari keyboard (Kurniawati, 2010).

Untuk dapat merekam segala aktifitas keyboard, keylogger harus dijalankan setelah sistem operasi berjalan dan masuk pada tampilan desktop. Karena

keylogger merupakan perangkat lunak yang bersifat memantau maka *keylogger* akan dijalankan secara *background process* dan tidak diketahui oleh pengguna yang menggunakan komputer tersebut.

XML Web Service

Web service merupakan perangkat lunak yang mendukung interoperabilitas dan interaksi antar sistem pada suatu jaringan. *Web service* ini akan memfasilitasi suatu sistem yang membutuhkan layanan untuk berkomunikasi dengan sistem yang lain (Ramadha, 2010).

Karena sistem yang akan dibangun bersifat *client server* maka dibutuhkan sebuah *web service* untuk dapat berkomunikasi dengan sistem yang lain. Informasi yang dikirim dari *client* ke *server* atau dari *server* ke *client* di format dalam bentuk XML sehingga data ini dapat diakses oleh sistem walaupun berbeda *platform*.

Sedangkan untuk proses pengiriman atau penerimaan data binary image dari *client* ke *server* atau dari *server* ke *client* tetap menggunakan format XML *web service* sebagai mediana, namun dibutuhkan suatu pengkodean yang mampu mengubah format *binary image* menjadi *ASCII string* yaitu menggunakan format *Base64 encode* dan *Base64 Decode*, metode ini mampu mngubah data yang semula berupa *binary image* menjadi *ASCII string*.

Android C2DM

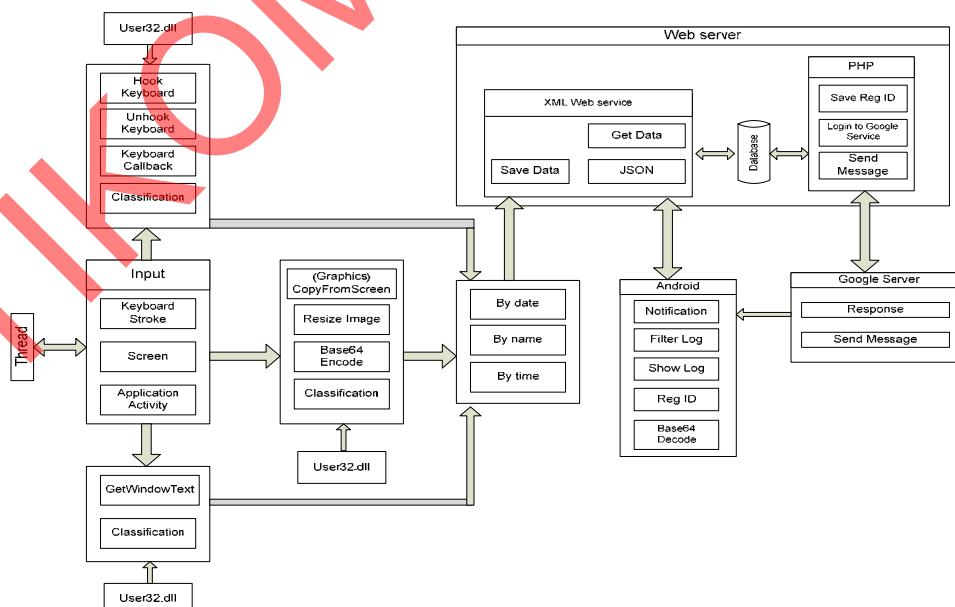
Semenjak *android* versi 2.2 yang dinamakan *Froyo* diluncurkan, *Google* telah menambahkan layanan baru untuk membantu pengembang untuk dapat mengirim data dari *server* ke aplikasi mereka pada ponsel *android*. *Android Cloud To Device Messaging*

(*C2DM*) membuat lebih mudah aplikasi *mobile* untuk melakukan sinkronisasi data dengan *server* (Huang, 2010).

Sebelum dapat menggunakan layanan *C2DM* aplikasi harus didaftarkan terlebih dahulu pada *server Google* menggunakan nama paket aplikasi misalnya *com.project.ispy* dan menyertakan email *Google* yang masih aktif untuk diverifikasi oleh pihak *Google*.

Sedangkan mekanisme pengiriman pesan melalui *C2DM*, pertama-tama aplikasi *android* melakukan koneksi ke *C2DM Fontend* menggunakan email yang sudah diverifikasi untuk mendapatkan registrasi id, setiap perangkat akan mendapatkan registrasi id yang berbeda-beda, namun registrasi id tersebut bersifat tidak permanen oleh karena itu sistem akan memperbaharui registrasi id secara berkala. Selanjutnya aplikasi *android* harus mengirimkan registrasi id tersebut pada *web server* untuk disimpan tujuannya adalah sebagai asumsi saat pengiriman data agar tidak terjadi kesalahan pada penerima. Pengiriman pesan dari *web server* ke *server C2DM* dimulai dari *web server* melakukan login pada *server Google* melalui *Client Login Service* dan jika berhasil maka akan mendapatkan *Auth token* sementara yang unik, *Auth Token* dan registrasi id yang disimpan tersebut disertakan pada saat mengirim pesan dari *web server* ke *C2DM Frontend*. Permintaan yang sudah di autentikasi selanjutnya akan dimasukkan kedalam *C2DM Message Queue* yang nantinya diteruskan ke aplikasi *mobile* menggunakan koneksi *TCP/IP*.

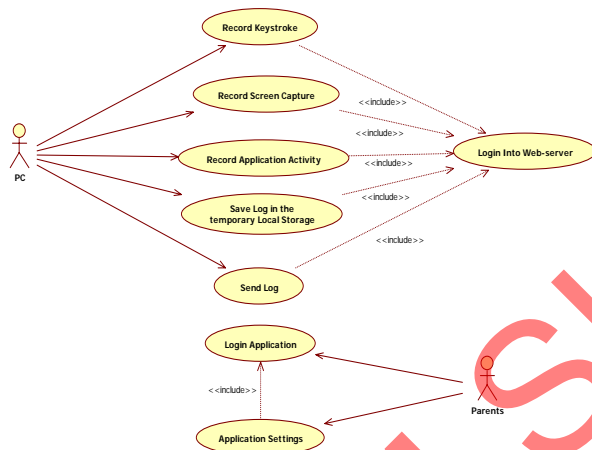
Gambaran Umum Sistem



Gambar 1 Blok diagram rancangan sistem

Pada gambar 1 sistem yang dibangun membutuhkan tiga buah aplikasi yang mendukung *server side* dan *client side*. Aplikasi yang berjalan di sisi *server side* berupa *web server* dan *web service* yang berfungsi menyimpan data *log* atau *history* dan penghubung antara aplikasi *client side*. Sedangkan aplikasi *client side* terdiri dari aplikasi *keylogger* yang diinstall pada komputer yang akan diintai aktifitasnya dan aplikasi *mobile* yang digunakan untuk *smartphone android* sebagai media penerima informasi *log*. Agar dapat melakukan fitur dari sistem ini semua perangkat harus terhubung dengan internet terutama aplikasi *keylogger*, karena berdasarkan interval yang sudah ditentukan *keylogger* akan secara otomatis mengirimkan *log* atau *history* ke *web server* untuk disimpan.

Use Case Diagram

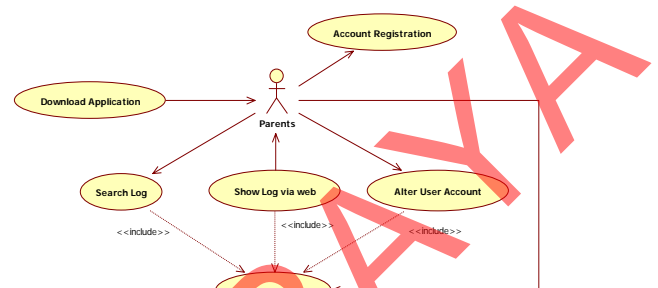


Gambar 2 Use Case Diagram Aplikasi Keylogger

Tabel 1 Penjelasan Singkat *Use Case Diagram* aplikasi Desktop *Keylogger*

Nama Use Case	Deskripsi
Login Application	Proses yang menangani login pada aplikasi <i>keylogger</i> sebelum dapat melakukan pengaturan aplikasi <i>keylogger</i> .
Application Setting	Proses yang menangani pengaturan aplikasi <i>keylogger</i> seperti interval <i>capture</i> dan interval <i>log</i> .
Login into Web-server	Proses yang menangani login ke server sebelum dapat melakukan pengiriman <i>log</i> ke server.
Record Keystroke	Proses yang merekam segala aktivitas keyboard
Record Screen Capture	Proses yang <i>capture</i> aplikasi yang berjalan
Record Application Activity	Proses yang merekam segala aktivitas aplikasi

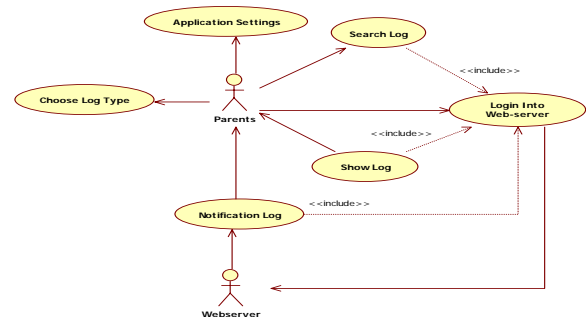
Nama Use Case	Deskripsi
Save log in the Temporary Local Storage	Proses yang digunakan untuk menyimpan sementara <i>log</i> pada penyimpanan lokal sebelum dikirim ke server
Send Log	Proses yang menangani pengiriman <i>log</i> ke server via internet.



Gambar 3 Use Case Diagram Aplikasi Website

Tabel 2 Penjelasan Singkat *Use Case Diagram* Aplikasi WebSite

Nama Use Case	Deskripsi
Account Registration	Proses yang menangani pendaftaran pengguna baru yang akan menggunakan sistem ini
Search Log	Proses yang menangani Pencarian Log
Download Application	Proses yang digunakan untuk melakukan unduh aplikasi dari web site
Login	Proses yang menangani login pengguna sebelum masuk pada halaman <i>member area</i> .
Alter User Account	Proses yang menangani pengaturan pengguna akun.
Show Log Via Web	Proses yang menangani penampilan log via web site.



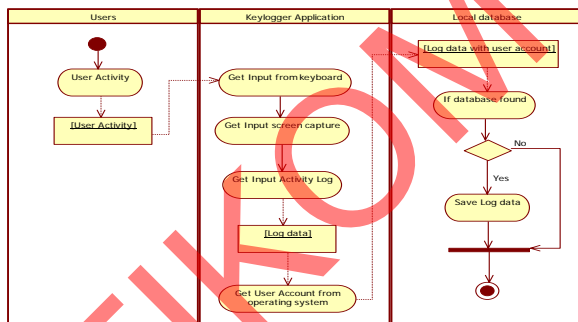
Gambar 4 Use Case Diagram Aplikasi Android

Tabel 3 Penjelasan Singkat *Use Case Diagram* Aplikasi Android

Nama Use Case	Deskripsi
Login Into Web server	Proses yang menangani login ke Web server sebelum dapat menerima log
Application Settings	Proses yang menangani pengaturan aplikasi
Show Log	Proses yang digunakan untuk menampilkan log
Search Log	Proses yang digunakan untuk mencari log
Choose Log	Proses yang digunakan untuk memilih tipe log yang digunakan
Notification Log	Proses yang dijalankan ketika ada pemberitahuan log baru

Activity Diagram

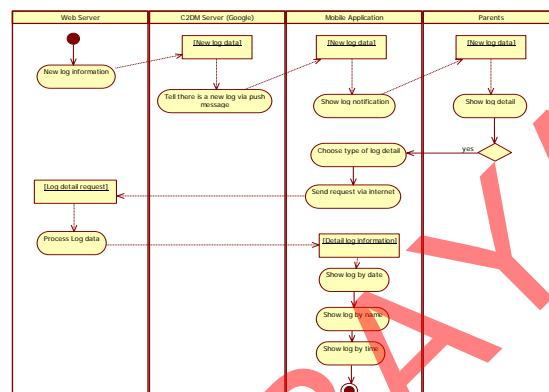
Proses merekam aktifitas *keyboard* dimulai saat ada interaksi dari pengguna yang memakai komputer atau disebut dengan *user activity*, *user activity* bisa berupa aktivitas yang terjadi saat *keyboard* komputer ditekan dan berupa *screen capture* dari layar aplikasi yang sedang aktif hasilnya berupa data *log*, proses ini dilakukan secara otomatis oleh aplikasi. Selanjutnya aplikasi akan menambahkan informasi pengguna yang menggunakan komputer tersebut, data ini diambil dari *user account* masing-masing komputer. Selanjutnya data data log akan disimpan sementara pada lokal database sebelum dikirim ke *web server*. Proses berikutnya jika *database* lokal ditemukan maka data *log* akan disimpan sementara pada tabel *keystroke*, namun jika *database* lokal tidak ditemukan maka data *log* tidak akan disimpan. Untuk lebih jelasnya dapat dilihat pada gambar 5.



Gambar 5 Activity Diagram Merekam Aktifitas Keyboard

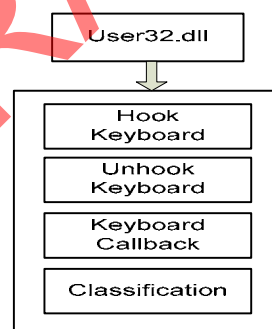
Proses menampilkan *log* dimulai dari dari *web server* yang telah menerima *log* baru dari aplikasi *keylogger* akan memberitahukan pada *server C2DM* bahwa ada *log* baru yang harus diberitahukan pada ponsel *android*, ponsel *android* yang menerima pesan tersebut akan menampilkan notifikasi *log* baru pada *parents*, informasi *log* baru tersebut bukan keseluruhan *log* melainkan hanya *header log* berupa tanggal dan pengguna komputer. *Parents* yang ingin melihat lebih detail lagi informasi *log* dapat memilih menu *preview*

log pada aplikasi *mobile*, selanjutnya permintaan akan dikirim ke *web server* menggunakan media internet, *web server* akan segera memproses permintaan dan menghasilkan informasi detail *log* dan mengirimkannya kembali pada ponsel. Untuk lebih jelasnya dapat dilihat pada gambar 6.



Gambar 6 Activity Diagram Menampilkan Log

HASIL DAN PEMBAHASAN Implementasi Keylogger



Gambar 7 Diagram Keylogger

Pada gambar 7 proses ini dilakukan ketika pengguna menekan setiap tombol di *keyboard*, kejadian ini ditangkap menggunakan fungsi *HookKeyboard()* yang merupakan *Windows API* dari sistem operasi *windows*, oleh karena itu proses ini memerlukan *library user32.dll* yang sudah disediakan *windows*.

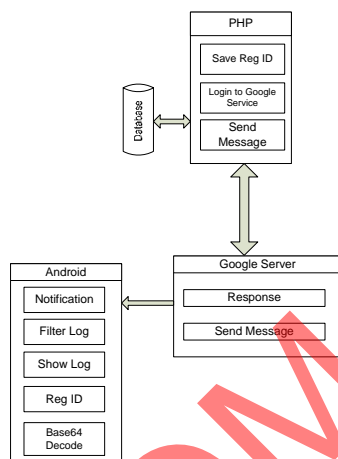
```
'create the hook
Public Sub HookKeyboard()
    callback = New
    KeyboardHookDelegate(AddressOf
    KeyboardCallback)
    KeyboardHandle =
    SetWindowsHookEx(13, callback,
    Process.GetCurrentProcess.MainModule
    .BaseAddress, 0)
End Sub
```

Selanjutnya fungsi *HookKeyboard()* akan memanggil setiap alamat pada tombol *keyboard* yang

sebelumnya berbentuk karakter ASCII Hex untuk diubah menjadi ASCII karakter, karena ASCII Hex berupa angka dan sulit untuk dibaca. Selanjutnya hasil dari fungsi *HookKeyboard()* akan diklasifikasikan berdasarkan nama *user account*, tanggal, waktu, judul aplikasi yaitu judul yang terdapat pada setiap aplikasi yang berjalan dan *log* dari inputan *keyboard* yang sudah berbentuk karakter string. Proses merekam aktifitas *keyboard* dapat dihentikan dengan cara memanggil fungsi *UnhookKeyboard()*.

```
Public Sub UnhookKeyboard()
    If (Hooked()) Then
        If
            UnhookWindowsHookEx(KeyboardHandle)
            <> 0 Then
                KeyboardHandle = 0
            End If
        End If
    End Sub
```

Implementasi C2DM



Gambar 8 Diagram Keylogger

Pada gambar 8 proses pengiriman pesan ke *android* menggunakan *push message* harus melewati *Google C2DM Server* terlebih dahulu. Oleh karena itu diperlukan sebuah *server* khusus yang menangani proses pengiriman pesan, *web server* harus mampu melakukan login ke *server C2DM* untuk mendapatkan token autentikasi. Alamat login ke *server google* sudah disediakan seperti berikut

```
curl
https://www.google.com/accounts/ClientLogin -d
Email=theEmailYouWhitelisted
-d Passwd=pass***word -d
accountType=HOSTED_OR_GOOGLE
-d source=Google-cURL-
Example -d service=ac2dm
```

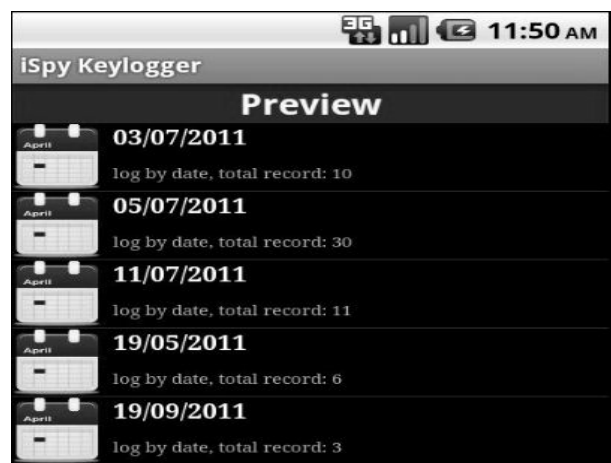
Setelah mendapatkan token autentikasi maka *web server* mendapatkan izin untuk mengirim pesan melalui *push message*. Namun pesan yang dikirim menggunakan *push message* hanya terbatas pada 1024 karakter.

```
url --header "Authorization:
GoogleLogin
auth=**authFromRegistrationAbov
e**"
"https://android.apis.google.co
m/c2dm/send" -d
registration_id=**phoneRegistra
tionId(reciever)** -d
"data.message=StringToPass" -d
collapse_key=something -k
```

Selain informasi berupa notifikasi, sistem juga mempunyai fungsi pencarian *log* agar pengguna juga dapat mencari *log* berdasarkan tanggal, nama pengguna dan waktu. Namun proses pencarian *log* ini sudah tidak lagi menggunakan *C2DM* sebagai pengirim pesan, sistem menggunakan protokol *http://* melalui internet untuk me-request dan menerima data dari *web server* dalam format XML dan JSON.

Untuk dapat menjalankan sistem ini dengan baik dan *realtime* semua perangkat harus terhubung dengan internet terutama aplikasi *keylogger*, karena berdasarkan interval yang sudah ditentukan *keylogger* akan secara otomatis mengirimkan *log* atau *history* ke *web server* untuk disimpan dan selanjutnya *web server* akan menghubungi *C2DM server* untuk mengirim pesan pada *smartphone android* secara *push message*.

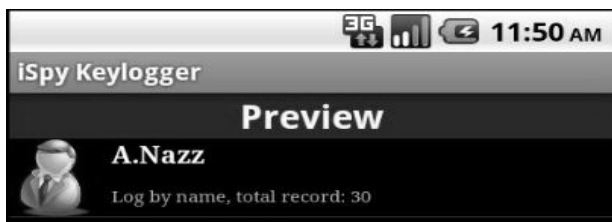
Pada gambar 10 berikut adalah *log* yang diklasifikasikan berdasarkan tanggal, dan juga menampilkan total *record* yang ada per tanggalnya.



Gambar 10 Klasifikasi Log Berdasarkan Tanggal

Pada gambar 11 berikut adalah hasil *log* yang diklasifikasikan berdasarkan nama pengguna komputer

yang diintai aktifitasnya dan juga menampilkan total record.



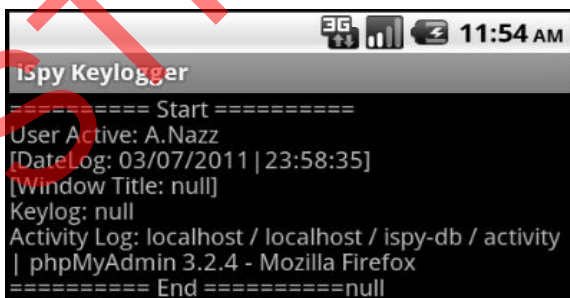
Gambar 11 Klasifikasi Log Berdasarkan Nama

Pada gambar 12 berikut adalah hasil log yang diklasifikasikan berdasarkan waktu, terdapat pembagian waktu yaitu pagi, siang, sore dan malam. Pagi dimulai dari pukul 00.00–05.59, siang dimulai pukul 06.00–11.59, sore dimulai pukul 12.00–17.59 dan malam dimulai pukul 18.00–23.59.



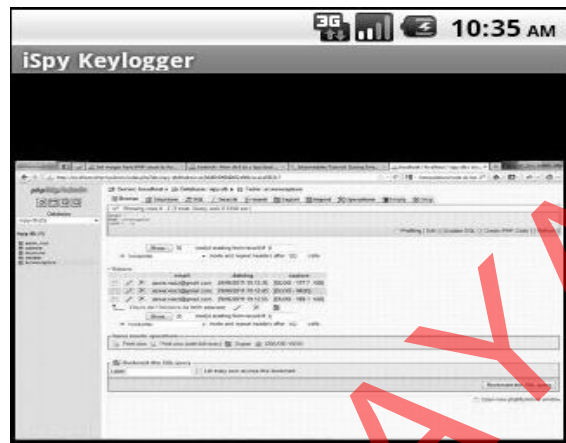
Gambar 12 Klasifikasi Log Berdasarkan Waktu

Pada gambar 13 berikut adalah hasil log yang muncul pada ponsel yaitu menampilkan informasi nama pengguna, tanggal, window title, keylog dan activity log nya



Gambar 13 Hasil Log pada Ponsel

Pada gambar 14 berikut adalah hasil log berupa screen capture, log capture tersebut dapat diperbesar atau di rotasi untuk memperjelas gambarnya.



Gambar 14 Hasil Log pada Ponsel berupa Screen Capture

SIMPULAN

Berdasarkan implementasi dan evaluasi yang telah dilakukan, dapat disimpulkan bahwa :

- Dengan layanan *push message* Android C2DM proses pengawasan aktifitas penggunaan komputer dapat dilakukan secara *realtime*. Dengan adanya sistem ini, peran orang tua untuk memantau aktifitas penggunaan komputer oleh anaknya tidak sampai terlewatkan.
- Proses integrasi aplikasi *keylogger*, *web server* dan *mobile android* berhasil dilakukan dengan melalui jaringan komunikasi internet.

RUJUKAN

- [1] Debel, Matthew, 2003, *National Center for Education Statistic 2003*, Institute Of Education Science.
- [2] Fowler, Martin, 2004, *UML Distilled Edisi 3 Panduan Singkat Bahasa Pemodelan Objek Standar*, ANDI, Yogyakarta.
- [3] Huang, Wei, 2010, *Android Cloud To Device Messaging* (Online), <http://android-developers.blogspot.com/2010/05/android-cloud-to-device-messaging.html> diakses 04-05-2011
- [4] Kurniawati, Dewi, 2010, *Kegunaan Keylogger*, Jurnal Teknologi Informasi.
- [5] Ramadha, Kartika, 2010, *Web Service* (Online) http://kartikanurramadha.info/a/web_service.pdf diakses tanggal 19-07-2011
- [6] Wibowo, Arip, 2011, *Web Server* (Online), <http://unyildadkan.com/internet/web-server/> diakses tanggal 06-05-2011