

Student Monitoring System pada J2ME menggunakan Web Service **(Studi Kasus: SMK Telekomunikasi Tunas Harapan Tenggara)**

¹⁾ Dewi Puspitasari, ²⁾ Hendry, ³⁾ Ramos Somya
Fakultas Teknologi Informasi
Universitas Kristen Satya Wacana
Jl. Diponegoro 52-60, Salatiga 50711, Indonesia
¹⁾ dewii.puspitasarii@yahoo.com, ²⁾ hendry@staff.uksw.edu
³⁾ ramos.6005@gmail.com

Abstract

Violations of the students is often become problems at school. As truant, fighting, smoking and others. In addition to teachers, parents also have an important role to contribute and supervise their children at school. Student monitoring system is important to help parents supervise their children at school. Both in value, attendance, payment data and the violation of the student in school. To facilitate parents in accessing this system, the system was built based on J2ME. J2ME is a development environment designed to put the Java software on mobile devices. The system can run on mobile devices that support Java applications. Server is built using web-based web service. Web services allow to exchange the use of existing functions so can use the functions from one application to make an application to another on a different network. That way, parents can get information at any time without limited by the number of characters.

Keywords: *Student Monitoring, J2ME, Web Service*

Abstrak

Pelanggaran siswa sering menjadi masalah di sekolah. Seperti membolos, berkelahi, merokok dan lain-lain. Selain guru, orang tua juga punya peran penting untuk turut serta mengawasi anaknya di sekolah. Sistem *monitoring* siswa membantu orang tua mengawasi anaknya di sekolah baik dalam nilai, absensi, data pembayaran maupun pelanggaran yang dilakukan siswa di sekolah. Untuk memudahkan orang tua dalam mengakses sistem ini, maka dibangun sistem yang berbasis J2ME. J2ME adalah lingkungan pengembangan yang didesain untuk meletakkan perangkat lunak Java pada perangkat selular. Sistem ini dapat berjalan pada perangkat selular yang mendukung aplikasi java. *Server* dibangun berbasis *web* dengan memanfaatkan *web service*. *Web service* memungkinkan untuk melakukan pertukaran pemakaian fungsi-fungsi yang ada sehingga bisa memakai fungsi dari aplikasi yang satu untuk membuat suatu aplikasi yang lain pada jaringan yang berbeda. Dengan begitu maka orang tua dapat memperoleh informasi kapanpun tanpa terbatas oleh jumlah karakter.

Kata Kunci: *Monitoring Siswa, J2ME, Web Service*

1. Pendahuluan

Ketertiban siswa sering kali menjadi suatu masalah di sebuah sekolah, apalagi pada jenjang sekolah menengah atas. Banyak kenakalan siswa yang terjadi di sekolah, seperti membolos sekolah, penyelewengan uang sekolah, terlibat tawuran, dan lain-lain. Walaupun pihak sekolah mempunyai bagian kesiswaan yang bertugas untuk memberi sanksi tegas terhadap siswa yang melanggar peraturan di sekolah, serta wali kelas yang selalu memantau nilai dan sikap siswa di sekolah, namun sering kali tidak efektif dan mengalami hambatan karena tidak didukung oleh guru-guru yang lainnya dan kurangnya kepedulian terhadap siswa. Oleh karena itu peran aktif dari orang tua juga sangat penting. Orang tua harus bisa ikut mengawasi segala bentuk kegiatan anaknya di sekolah. Untuk memudahkan orang tua mengawasi anaknya di sekolah, diperlukan suatu sistem *monitoring* yang mampu menyediakan informasi bagi orang tua mengenai absensi, nilai, data pelanggaran dan data keuangan.

Java 2 Micro Edition atau yang disebut J2ME adalah *platform* yang dapat berjalan pada sistem operasi Symbian. J2ME adalah lingkungan pengembangan yang didesain untuk meletakkan perangkat lunak Java pada barang elektronik beserta perangkat pendukungnya. J2ME biasa digunakan pada telepon selular, *pager*, *personal digital assistants* (PDA) dan sejenisnya [2].

Sekolah Menengah Kejuruan (SMK) Telekomunikasi Tunas Harapan adalah sekolah yang siswanya didominasi oleh laki-laki. Rata-rata siswanya berasal dari luar kota. Sehingga mereka harus tinggal di asrama atau kos-kosan [3]. Mereka jauh dari orang tua dan perilaku mereka tidak diawasi. Banyak pengaruh buruk dalam pergaulan yang diberikan. Daftar pelanggaran yang tercatat juga meningkat dari tahun sebelumnya. Pertengahan tahun 2011 tercatat 67 kasus pelanggaran yang dilakukan siswa. Dan pertengahan tahun 2012 tercatat 82 pelanggaran yang dilakukan oleh siswa [4]. Bukan tidak mungkin pelanggaran ini akan terus meningkat pada akhir tahun 2012. Secara garis besar banyaknya pelanggaran yang dilakukan oleh siswa akan berpengaruh terhadap kemajuan dan prestasi belajar di sekolah [5]. Sistem *monitoring* yang ada di SMK Telekomunikasi Tunas Harapan saat ini masih menggunakan teknologi *Short Message Service* (SMS), sehingga orang tua harus berulang kali mengirimkan SMS kepada wali kelas untuk memantau anaknya di sekolah. Informasi yang di dapatkan juga terbatas oleh panjang karakter SMS. Dari hasil kuisisioner yang dibagikan, didapat kesimpulan bahwa sebagian besar orang tua siswa di SMK Telekomunikasi Tunas Harapan menggunakan perangkat *mobile* yang mendukung teknologi Java.

Melihat permasalahan-permasalahan tersebut maka munculah keinginan untuk membangun sebuah *Student Monitoring System* pada J2ME menggunakan *Web Service*. *Web service* memiliki keunggulan dalam pertukaran data antara *server* dan *klien*.

2. Tinjauan Pustaka

Salah satu penelitian yang membahas tentang sistem *monitoring* adalah “Sistem *Monitoring* Presensi dan Akademik Siswa Berbasis SMS Gateway (Studi Kasus: SMA Persatuan Tulangan)”. Sistem *monitoring* ini menyediakan informasi mengenai nilai, presensi, kegiatan sekolah dan catatan dari wali kelas. Pihak sekolah akan mengirimkan SMS kepada orang tua tentang presensi hariannya. Sistem *monitoring* yang dibangun menggunakan *SMS Gateway* sebagai *monitoring* bagi pihak orang tua. Sedangkan sistem *monitoring* bagi pihak sekolah menggunakan *web* [6].

Penerapan *web service* dalam pengembangan sistem sudah cukup banyak digunakan. Pada penelitian sebelumnya yang berjudul “Perancangan *Prototype* Aplikasi *Mobile* Untuk Pengaksesan *Web Service*” membahas perancangan *prototype* aplikasi *mobile* menggunakan *Java Micro Edition* (Java ME) untuk mengakses *web service*. *Web service* yang diakses oleh aplikasi yang dirancang adalah *web service* pada suatu Sistem Informasi Akademik. *Web service* ini mengembalikan Indeks Prestasi Semester, Indeks Prestasi Kumulatif, dan Nilai suatu Mata kuliah. Hasil pengujian terhadap *prototype* aplikasi menunjukkan bahwa aplikasi bekerja sesuai dengan yang diharapkan [7].

Penelitian-penelitian yang sudah ada dijadikan landasan dalam perancangan *Student Monitoring System* Pada J2ME Dengan *Web Service* ini. Perbedaan dengan penelitian sebelumnya yang dilakukan oleh Moch. Awang Putra yaitu sistem yang akan dibangun menggunakan teknologi *web service* yang memberikan kemudahan dalam pertukaran data. Informasi yang dihasilkan pada penelitian sebelumnya terbatas oleh panjang karakter SMS dan menghabiskan banyak biaya karena orang tua harus berkali-kali mengirim SMS untuk mendapat informasi.

Student Monitoring

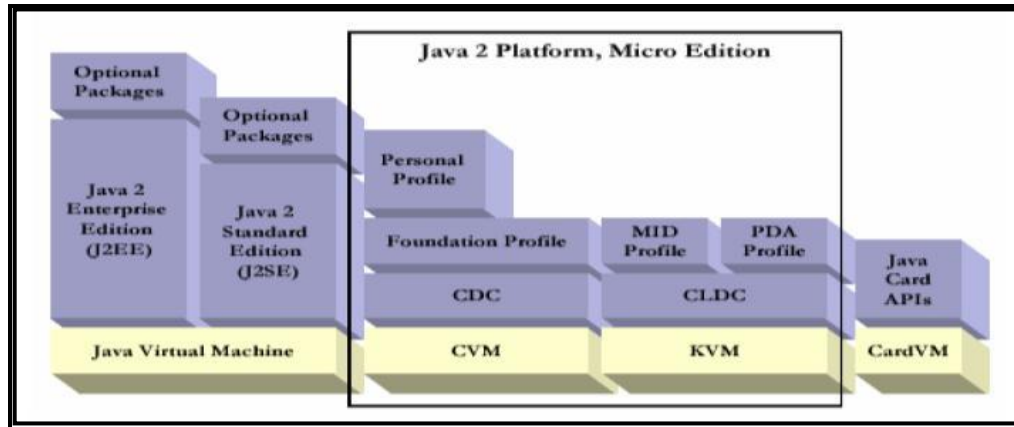
Monitoring adalah proses rutin pengumpulan data dan pengukuran kemajuan atas objektif program. Memantau perubahan, yang fokus pada proses dan keluaran. *Monitoring* melibatkan perhitungan atas apa yang dilakukan. *Monitoring* melibatkan pengamatan atas kualitas dari layanan yang diberikan. *Student Monitoring* adalah melakukan pengawasan dan pengontrolan terhadap perkembangan akademik siswa di sekolah. Baik itu dalam proses nilai siswa, pelanggaran ataupun proses ketidakhadiran siswa dalam proses kegiatan belajar-mengajar [8].

J2ME

Java 2 Micro Edition (J2ME) adalah lingkungan pengembangan yang didesain untuk meletakkan perangkat lunak java pada barang elektronik beserta perangkat pendukungnya. Pada J2ME, jika perangkat lunak berfungsi baik pada sebuah perangkat maka belum tentu juga berfungsi baik pada perangkat yang lainnya. J2ME membawa java ke dunia informasi, komunikasi, dan

perangkat komputasi selain perangkat komputer *desktop* yang biasanya lebih kecil dibandingkan perangkat komputer *desktop*. J2ME biasa digunakan pada telepon selular, *pager*, *personal digital assistants* (PDA) dan sejenisnya [3].

J2ME adalah bagian dari J2SE, karena itu tidak semua *library* yang ada pada J2SE dapat digunakan pada J2ME. Tetapi J2ME mempunyai beberapa *library* khusus yang tidak dimiliki J2SE. Arsitektur J2ME dapat dilihat pada Gambar 1 :



Gambar 1 Arsitektur J2ME [9]

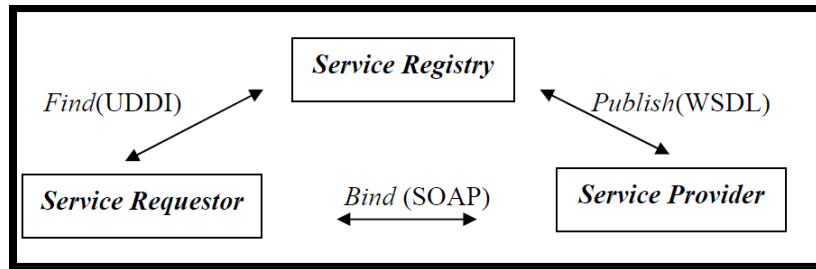
Dari Gambar 1, J2ME mempunyai lapisan konfigurasi dan *profile* yang didukung oleh *Java Virtual Machine* (*C-Virtual Machine* dan *K-Virtual Machine*). Aplikasi yang berjalan pada sebuah perangkat yang mendukung MIDP disebut dengan MIDlet. Dengan kata lain MIDlet merupakan aplikasi dari MIDP (*Mobile Information Device Profile*). MIDlet berupa kelas abstrak yang merupakan sub kelas dari bentuk dasar aplikasi sehingga antarmuka antar aplikasi J2ME dan aplikasi manajemen pada perangkat dapat terbentuk [10].

Web Service

Web Service adalah sekumpulan *application logic* beserta *object-object* dan *method-method* yang dimilikinya yang terletak di suatu *server* yang terhubung ke internet sehingga dapat diakses menggunakan *protocol* HTTP dan SOAP. Tujuan dari teknologi ini adalah untuk memudahkan beberapa aplikasi atau komponennya untuk saling berhubungan dengan aplikasi lain dalam sebuah organisasi maupun diluar organisasi menggunakan standar yang tidak terikat *platform* dan tidak terikat akan bahasa pemrograman yang digunakan. *Web service* menyimpan data informasi dalam format XML, sehingga data ini dapat diakses oleh sistem lain walaupun berbeda *platform*, sistem operasi maupun bahasa *compiler*. Penggunaan *Web Service* juga menawarkan kelebihan dan fleksibilitas, diantaranya [11] :

- Lintas *Platform*
- Jembatan Penghubung Dengan *Database*
- *Language Independent*
- Mempermudah Proses Pertukaran Data
- Penggunaan Kembali Komponen Aplikasi

Web Service mempublikasikan aplikasi dalam bentuk layanan (*services*) yang sekali dipublikasikan di internet akan bisa ditemukan dan diakses oleh aplikasi lain di internet tidak tergantung pada *platform* yang digunakan. Operasi suatu *web service* digambarkan sebagai hubungan antara tiga peran yang berbeda yaitu *Service Provider*, *Service Registry* dan *Service Requestor* [12]. Hubungan ketiganya digambarkan pada Gambar 2.



Gambar 2 Hubungan Antara Peran Dalam *Web Service* (Kuswandi, 2007)

- *Service Provider*: Berfungsi untuk menyediakan layanan/ *service* dan mengolah sebuah *registry* agar layanan-layanan tersebut dapat tersedia.
- *Service Registry*: Berfungsi sebagai lokasi central yang mendeskripsikan semua layanan/ *service* yang telah di-register.
- *Service Requestor*: Peminta layanan yang mencari dan menemukan layanan yang dibutuhkan serta menggunakan layanan tersebut.

SOAP

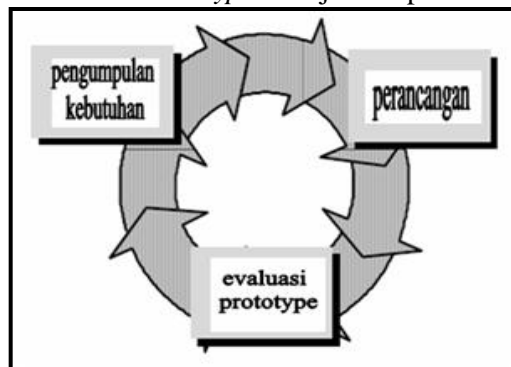
SOAP merupakan protokol yang digunakan untuk mempertukarkan data atau informasi dalam *format* XML. SOAP dapat dikatakan sebagai gabungan antara HTTP dengan XML karena SOAP umumnya menggunakan *protocol* HTTP sebagai sarana transportasi datanya dan data akan dipertukarkan ditulis dalam *format* XML. Karena SOAP menggunakan HTTP dan XML maka SOAP memungkinkan pihak-pihak yang mempunyai *platform*, sistem operasi dan perangkat lunak yang berbeda dapat saling mempertukarkan datanya [13]. Dalam J2ME, SOAP didefinisikan dalam JSR 172. JSR 172 adalah *package* standar akses J2ME ke *web service* [14]. JSR 172 mendefinisikan API untuk *Web Services*.

NuSoap

NuSOAP merupakan salah satu dari sekian *toolkit* yang tersedia untuk programmer PHP yang ingin bekerja dengan layanan SOAP. Keunggulan yang ditawarkan oleh NuSOAP adalah karena kesederhanaan sebab justru karena sederhana dan mudah sehingga kecepatannya menjadi lebih baik [15].

3. Metode Penelitian

Metode pengembangan sistem yang digunakan pada penelitian ini adalah metode *Prototyping*. *Prototyping* adalah proses yang digunakan untuk membantu pengembangan perangkat lunak dalam membentuk model dari perangkat lunak yang harus dibuat [16]. Metode ini dilakukan secara bertahap, yaitu dengan mengembangkan suatu *prototype* yang sederhana terlebih dahulu baru kemudian dikembangkan dari waktu ke waktu sampai perangkat lunak selesai dikembangkan. *Prototype* merupakan bentuk dasar atau model awal dari suatu sistem atau subsistem. Tahap-tahap dalam metode *Prototype* ditunjukkan pada Gambar 3.



Gambar 3 Tahapan dalam Model *Prototyping* [16]

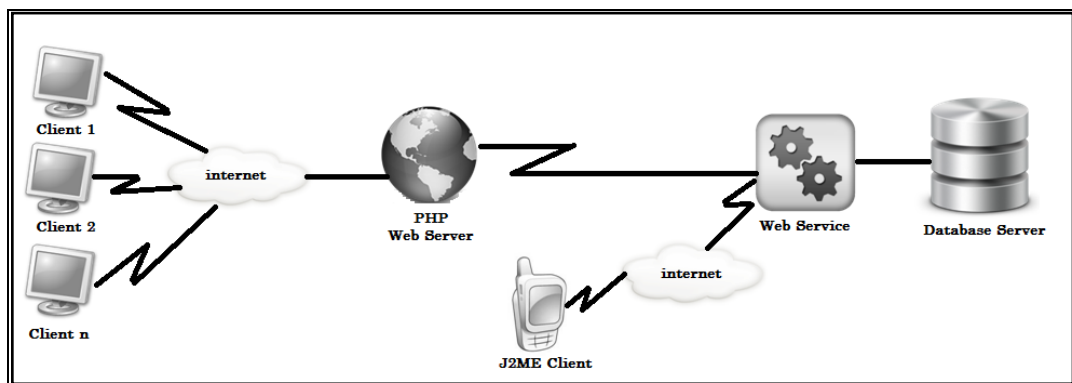
- Pengumpulan Kebutuhan

Tahap ini untuk mengetahui dan menerjemahkan semua permasalahan serta kebutuhan perangkat lunak dan kebutuhan sistem yang dibangun. Analisis kebutuhan dilakukan dengan melakukan pencarian data-data serta informasi-informasi yang dibutuhkan oleh sistem. Ada empat analisis kebutuhan dalam perancangan sistem yaitu analisis kebutuhan sistem, analisis kebutuhan data, analisis kebutuhan perangkat keras dan analisis kebutuhan perangkat lunak. Dalam tahap ini dilakukan analisis terhadap kebutuhan pengguna, kebutuhan pengguna tersebut antara lain:

- Orang Tua Siswa adalah sasaran utama dari pembuatan aplikasi. Diharapkan orang tua dapat memperoleh informasi mengenai absensi, nilai, pelanggaran dan data keuangan siswa di sekolah. Orang tua harus *login* dahulu agar bisa mengakses aplikasi ini dengan *username* dan *password* yang sudah ditentukan dari sekolah.
- Wali Kelas dapat melihat semua data tentang siswa dan mengisi catatan-catatan yang perlu diperhatikan oleh orang tua.
- Guru Mata Pelajaran dapat mengisi nilai siswa serta mengisi absensi.
- Guru BP atau Kesiswaan dapat mengisi pelanggaran-pelanggaran yang dilakukan siswa di sekolah.
- Administrasi di Tata Usaha dapat mengisi kewajiban keuangan siswa yang harus dibayar dan jatuh tempo pembayaran.
- Admin dapat melakukan pengaturan penuh atas sistem. Seperti menambah, mengubah dan menghapus data siswa, melakukan pengaturan hak akses *user*, serta menambah, mengubah atau menghapus operator pengguna sistem, menambah, mengubah serta menghapus data pelanggaran.

Perancangan Sistem

Setelah melakukan pengumpulan kebutuhan, selanjutnya yang dilakukan adalah melakukan perancangan arsitektur dan perancangan sistem. Gambar 4 menunjukkan rancangan arsitektur untuk *student monitoring system*.

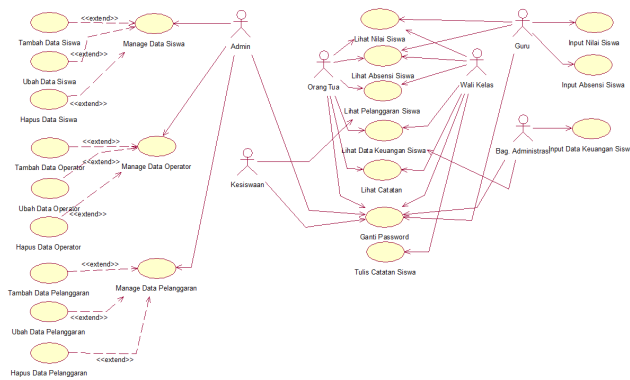


Gambar 4 Rancangan Arsitektur *Student Monitoring System*

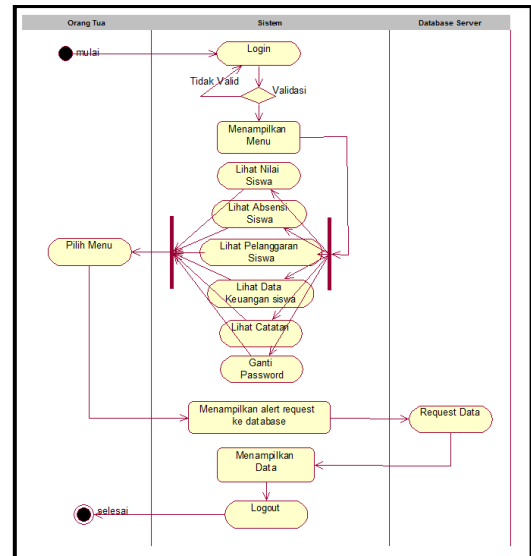
Gambar 2 merupakan rancangan arsitektur *student monitoring system*. Terdapat tiga aplikasi yaitu *web server*, *J2ME* dan *web service*. *Web service* menghubungkan antara *web server* dan *J2ME* dengan *database server*.

Perancangan sistem dibuat menggunakan diagram UML. Rancangan yang ada mewakili semua aspek *software* yang diketahui, dan rancangan ini sebagai dasar dalam pembuatan *prototype*.

Use case diagram menggambarkan interaksi antara aktor-aktor dengan sistem yang dibangun serta menggambarkan fungsionalitas yang dapat diberikan sistem kepada *user*. *Use case diagram* sistem ditunjukkan pada Gambar 5.



Gambar 5 Use case Diagram Student Monitoring System

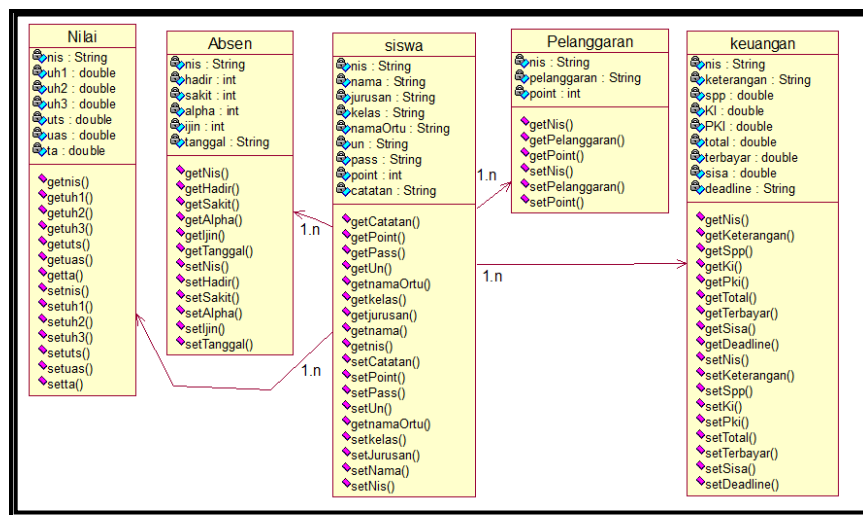


Gambar 6 Activity Diagram Orang Tua

Activity diagram menggambarkan aliran aktifitas dalam sistem yang sedang dirancang, bagaimana masing-masing alir berawal, *decision* yang mungkin terjadi, dan bagaimana mereka berakhir. Diagram mewakili aktivitas setiap aktor dalam menggunakan sistem. Gambar 6 merupakan *activity diagram* orang tua saat masuk ke sistem.

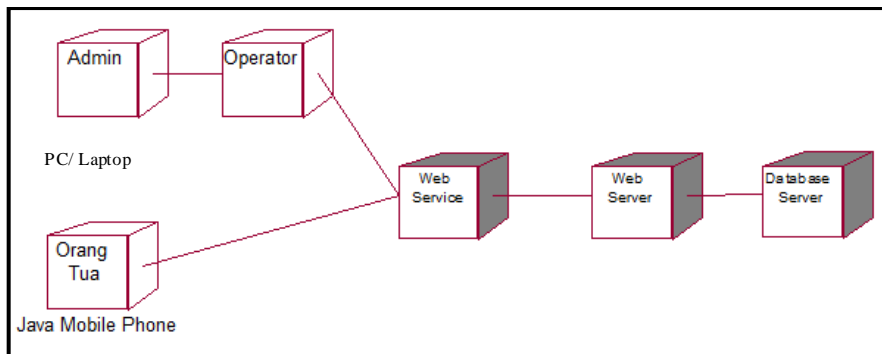
Dari Gambar 6 dapat dijelaskan, orang tua harus *login* dahulu sebelum masuk ke menu utama. Setelah *username* dan *password valid*, maka akan tampil menu utama. Terdapat 6 menu utama yaitu lihat nilai, lihat absensi, lihat pelanggaran, lihat data keuangan, lihat catatan dan ganti *password*. Setelah memilih salah satu menu, aplikasi akan menampilkan konfirmasi untuk meneruskan *request* ke *database*. Kemudian sistem akan menampilkan data.

Class diagram aplikasi *mobile* ditunjukkan Gambar 7.



Gambar 7 Class Diagram Aplikasi Mobile

Deployment Diagram adalah *Deployment/ physical* diagram menggambarkan detail bagaimana komponen di-*deploy* dalam infrastruktur sistem, di mana komponen akan terletak (pada mesin, *server* atau piranti keras apa), bagaimana kemampuan jaringan pada lokasi tersebut, spesifikasi *server*, dan hal-hal lain yang bersifat fisik.



Gambar 8 Deployment Diagram Student Monitoring System

Gambar 8 merupakan *deployment diagram* dari sistem *monitoring* yang dibuat. Terdapat beberapa perangkat yang digunakan saat *deployment* yaitu PC/ laptop yang digunakan untuk mengimplementasikan aplikasi *web*, serta perangkat *Java mobile* untuk mengimplementasikan aplikasi *mobile*. Saat orang tua *request* data maka aplikasi akan terhubung pada *web server*. Pertukaran data menggunakan *web service* sebagai jembatan penghubung.

- Evaluasi *Prototype*

Proses evaluasi *prototyping* dilakukan sebanyak dua kali pengujian dilakukan oleh *user*. Penjelasan setiap tahap evaluasi *prototyping* adalah sebagai berikut :

a. Evaluasi Tahap Pertama

Evaluasi *prototype* pertama dilakukan setelah perancangan awal aplikasi. Dalam tahap ini melakukan perancangan *database* dan desain tampilan. Hasil dari evaluasi tahap pertama antara lain: (1) Memperbaiki *database* sistem sesuai dengan yang dibutuhkan; (2) Memperbaiki tampilan supaya lebih menarik.

b. Evaluasi Tahap Kedua

Pada tahap kedua dilakukan penambahan proteksi sistem dan perbaikan tampilan aplikasi *mobile*. Proteksi yang dilakukan adalah *user* tidak dapat masuk ke menu utama apabila *username* atau *password* tidak sesuai. Pada halaman utama aplikasi *mobile* juga ditambah menu lupa *password*. Menu ini berfungsi untuk mengingatkan kembali *password* yang digunakan untuk *login* ke sistem.

4. Hasil Dan Pembahasan

Terdapat dua aplikasi untuk *student monitoring system* ini, yaitu aplikasi *mobile* dan aplikasi *web*. Kedua aplikasi dihubungkan oleh *library* nuSoap. Kode Program 1 merupakan kode program untuk koneksi menggunakan nuSoap.

Kode Program 1 Perintah Koneksi Dengan NuSoap

```

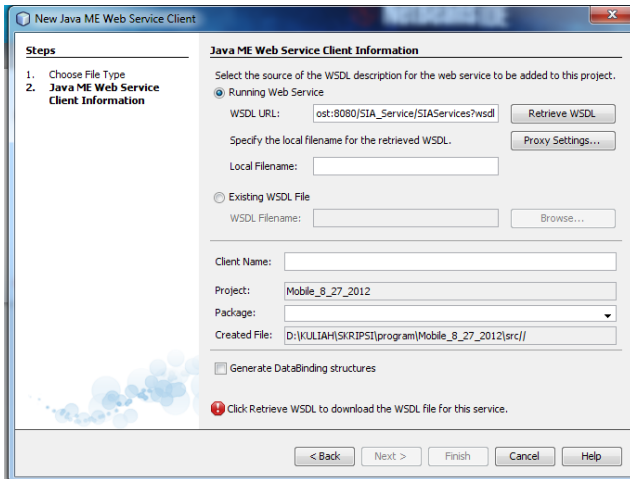
1. <?php
2. require_once("../nusoap/nusoap.php");
3. session_start();
4. $client = new
5. nusoap_client("http://localhost:8080/SIA_Service/
6. SIAServices?WSDL", true);
7. ?>

```

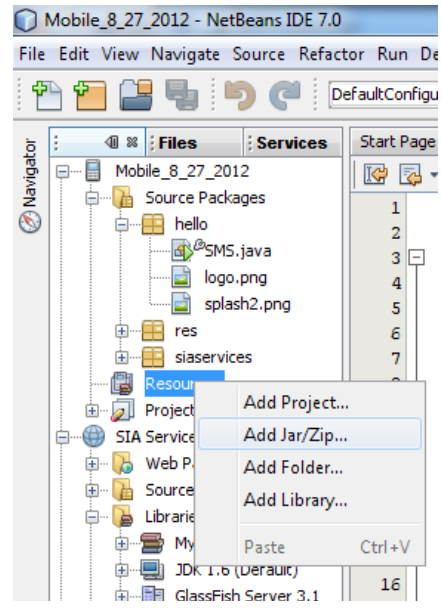
Kode Program 1 merupakan perintah koneksi menggunakan nuSoap pada aplikasi *web* yang melakukan *request* ke alamat WSDL dari *service* yang telah dibuat.

Berikut langkah untuk membuat koneksi antara *web service* dengan aplikasi *mobile*:

1. Klik kanan pada *project mobile* yang dibuat, dan pilih Java ME *Web Service Client*.
2. Kemudian tulis alamat WSDL ke WSDL URL. Gambar 9 merupakan tampilan ketika membuat Java ME *Web Service Client*.



Gambar 9 Membuat Koneksi Antara *Web Service* Dengan Aplikasi *Mobile*



Gambar 10 Menambah *Library JSR 172* ke Aplikasi *Mobile*

3. Pilih *button Retrieve WSDL*, kemudian pilih *button Next* dan *Finish*.

Kode Program 2 merupakan perintah untuk melakukan pemanggilan kelas *SIAService_Stub* pada *package siaservice*.

Kode Program 2 Perintah Pemanggilan Kelas *SIAService_Stub* pada *Package siaservice*

```
1. import siaservices.SIAServices_Stub;
```

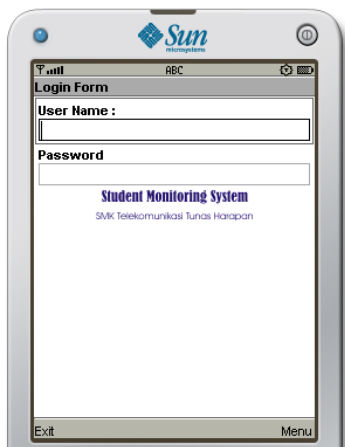
Untuk mendefinisikan API dari *web service* digunakan *library JSR 172*. Berikut adalah langkah untuk menambahkan *library JSR 172* ke aplikasi *mobile*:

1. Pilih *Resources* kemudian klik kanan dan pilih *Add Jar/ Zip*. Gambar 10 merupakan tampilan ketika menambah *library JSR 172*.
2. Kemudian pilih lokasi disimpannya *library JSR 172* dan pilih *open*.

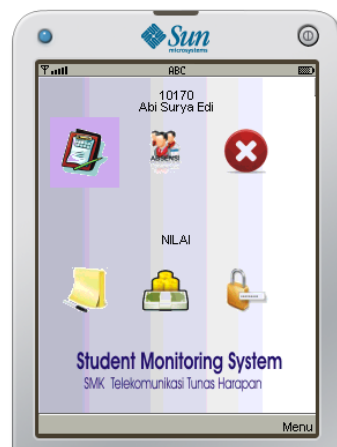
Kode Program 3 merupakan perintah untuk melakukan pemanggilan *library JSR 172*.

Kode Program 3 Perintah Pemanggilan *Library JSR 172*

```
1. import java.rmi.RemoteException;
```



Gambar 11 Halaman *Login* Pada Aplikasi *Mobile*



Gambar 12 Halaman *Menu Utama*

Antarmuka Aplikasi *Client*

Tampilan awal dari aplikasi *mobile* adalah *splashscreen*. Setelah beberapa saat akan muncul *form login*. Orang tua harus *login* untuk masuk ke menu utama. Pada *form login* terdapat pilihan *login* dan *lupa password*. Gambar 11 merupakan tampilan *form login* pada aplikasi *mobile*.

Kode program 4 merupakan kode dari *service login* pada aplikasi *mobile*.

Kode Program 4 Perintah Untuk Penanganan *Service Login* Pada Aplikasi *Mobile*

```

1. @WebMethod(operationName = "LoginSiswa")
2. public boolean LoginSiswa (String username,
3. String pass) {
4. boolean ret = false;
5. try {
6. this.conn = new MyConnection().getConnection();
7. PreparedStatement stm =
8. conn.prepareStatement("SELECT * FROM tsiswa
9. WHERE username = '" + username + "' and password = '" + pass + "'");
10. ResultSet rs = stm.executeQuery();
11. while (rs.next()) {
12. ret = true;
13. }
14. } catch (Exception e) {
15. }
16. return ret;
17. }

```

Kode program 5 merupakan kode program untuk *login* pada aplikasi *mobile*. Pertama dipanggil *service login* orang tua yang sudah dibuat, apabila *username* dan *password* benar maka aplikasi akan menampilkan menu utama. Apabila *username* atau *password* salah maka akan ditampilkan pesan gagal *login*.

Kode Program 5 Perintah Untuk *Login* Pada Aplikasi *Mobile*.

```

1. class WSClient implements Runnable {
2. public void run() {
3. try {
4. un = txUserName.getString();
5. pass = txPassword.getString();
6. st = new SIAServices_Stub();
7. boolean cekLogin = st.loginSiswa(un, pass);
8. if (cekLogin) {
9. nama = st.getNamaSiswa(un);
10. nis = st.getNis(un);
11. startKanvas();
12. } else {
13. ketLogin.setText("User Name / Password Salah");
14. txUserName.setString("");
15. txPassword.setString("");
16. }
17. } catch (Exception e) {
18. }
19. }
20. }

```

Setelah berhasil *login* maka akan muncul menu utama. Terdapat 6 menu utama yaitu nilai, absensi, pelanggaran, catatan, keuangan dan ganti *password*. Orang tua dapat memilih menu sesuai informasi yang ingin mereka dapatkan. Gambar 12 merupakan tampilan menu utama pada aplikasi *mobile*.

- Antarmuka Aplikasi *Web*

Tampilan awal dari aplikasi *web* adalah halaman *login*. Seperti pada perancangan, pada implementasi juga diterapkan demikian. Yaitu terdapat 5 pilihan *login* sesuai hak akses yang dimiliki setiap operator. Tampilan utama pada aplikasi *web* dapat ditunjukkan oleh Gambar 13.



Gambar 13 Halaman *Login* Pada Aplikasi *Web*

Kode program 6 merupakan kode program dari *service login* pada halaman *web*.

Kode Program 6 Perintah Untuk Penanganan *Service Login* Pada Halaman *Web*

```

1. @WebMethod(operationName = "Login")
2. public boolean Login(@WebParam(name = "username") String username,
   @WebParam(name = "password") String password,
3. @WebParam(name = "tipe") String tipellogin) {
4. boolean ret = false;
5. try {
6. this.conn = new MyConnection().getConnection();
7. PreparedStatement stm = conn.prepareStatement("SELECT * FROM tuser WHERE
   username = '" + username + "'");
8. ResultSet rs = stm.executeQuery();
9. while (rs.next()) {
10. String tipe = rs.getString(4);
11. String user = rs.getString(5);
12. String pass = rs.getString(6);
13. if (user.equals(username) && pass.equals(password) && tipe.equals(tipellogin)) {
14. ret = true;
15. }
16. break;
17. }
18. } catch (Exception e) {
19. }
20. return ret;
21. }

```

Kode program 7 merupakan kode program untuk *login* pada halaman *web*. Pertama dipanggil *service login* yang sudah dibuat, apabila *username* dan *password* benar maka sistem akan melakukan cek terhadap hak akses yang digunakan. Apabila hak akses adalah *admin* maka akan ditampilkan halaman *admin*. Apabila hak akses adalah guru maka akan ditampilkan halaman guru. Apabila hak akses adalah kesiswaan maka akan ditampilkan halaman kesiswaan. Apabila hak akses adalah wali kelas maka akan ditampilkan halaman wali kelas. Apabila hak akses adalah administrasi maka akan ditampilkan halaman administrasi. Apabila *username* atau *password* salah maka akan ditampilkan pesan gagal *login*.

Kode Program 7 Perintah Untuk *Login* Pada Halaman *Web*.

```

1. <?php
2. require_once("../nusoap/nusoap.php");
3. require_once("../kon/koneksi.php");
4. $param =
5. array("username"=>$_POST["txtid"],"password"=>$_POST["txtpass"],"tipe
6. =>$_POST["txthakakse
7. "]);
8. $ket = $_GET['ket'];
9. $hak = $_GET['id'];
10. $result = $client->call('Login',$param);
11. $cek = $result["return"];
12. if($cek == "true"){
13. $_SESSION['loginid'] = $_POST["txtid"];
14. $_SESSION['pass'] = $_POST["txtpass"];
15. $_SESSION['loginakses'] = $_POST["txthakakses"];
16. $hak = $_POST["txthakakses"];
17. if($hak == "ADMIN"){
18. echo "<script> document.location.href='../admin.html'; </script>";
19. }else if($hak == "KESISWAAN"){
20. echo "<script> document.location.href='../kesiswaan.html'; </script>";
21. }else if($hak == "GURU"){
22. echo "<script> document.location.href='../guru.html'; </script>";
23. }else if($hak == "WALIKELAS"){
24. echo "<script> document.location.href='../walikelas.html'; </script>";
25. }else if($hak == "ADMINISTRASI"){
26. echo "<script> document.location.href='../administrasi.html';
27. </script>";
28. }
29. }else{
30. echo "<script>
31. document.location.href='../login.php?id=".$hak."&ket=PASSWORD DAN
32. USERNAME SALAH'; </script>";
33. }
34. ?>
    
```

- Pengujian Sistem

Pengujian ini berfungsi untuk melihat sejauh mana aplikasi ini dapat berjalan dan untuk menemukan kesalahan yang mungkin terjadi. Tahap ini merupakan tahapan untuk menemukan beberapa kekurangan yang ada dalam sistem. Pengujian aplikasi ini seperti yang dijelaskan pada tahap sebelumnya, menggunakan dua teknik pengujian, yaitu:

a. Pengujian Alfa

Pengujian alfa merupakan pengujian aplikasi yang dilakukan oleh pembuat ataupun orang-orang yang terlibat di dalamnya. Pengujian alfa hanya untuk sirkulasi internal dan masalah (*error*) atau ketidaklengkapan yang terdapat dalam aplikasi dapat diduga sebelumnya [13]. Pengujian sistem ini menggunakan metode *blackbox*. Metode ujicoba *blackbox* memfokuskan pada keperluan fungsional dari software. Karna itu ujicoba *blackbox* memungkinkan pengembang software untuk membuat himpunan kondisi input yang akan melatih seluruh syarat-syarat fungsional suatu program.

- Pengujian Aplikasi Mobile

Pengujian *login* yaitu untuk mengecek apakah fungsional dari *form login* telah berjalan dengan baik atau belum. Dari pengujian yang dilakukan didapatkan kesimpulan bahwa *form login* pada aplikasi *mobile* telah berfungsi dengan baik. Tabel 1 merupakan pengujian *login* untuk aplikasi *mobile*.

Tabel 1 Pengujian *Black Box* Untuk *Form Login* Aplikasi *Mobile*

Aktivitas	Input	Output	Status Pengujian
<i>Login</i>	<i>Username</i> dan <i>password</i>	a. Jika berhasil maka masuk ke menu utama b. Jika <i>password</i> salah maka muncul peringatan <i>password</i> salah	<i>Valid</i>

Pengujian menampilkan data yaitu untuk mengecek apakah fungsional dari *form* menampilkan data telah berjalan dengan baik atau belum. Dari pengujian yang dilakukan didapatkan kesimpulan bahwa *form* menampilkan data pada aplikasi *mobile* telah berfungsi dengan baik. Tabel 2 merupakan pengujian untuk menampilkan data.

Tabel 2 Pengujian *Black Box* Untuk Menampilkan Data

Aktivitas	Input	Output	Status Pengujian
Menampilkan data nilai, absensi, pelanggaran, catatan dan keuangan	-	a. Jika terdapat data, maka data akan tampil b. Jika tidak ada datanya maka akan muncul peringatan tidak ada data	Valid

Pengujian *logout* yaitu untuk mengecek apakah fungsional dari *form logout* telah berjalan dengan baik atau belum. Dari pengujian yang dilakukan didapatkan kesimpulan bahwa *form logout* pada aplikasi *mobile* telah berfungsi dengan baik. Tabel 3 merupakan pengujian *logout* untuk aplikasi *mobile*.

Tabel 3 Pengujian *Black Box* Untuk *Form Logout*

Aktivitas	Input	Output	Status Pengujian
<i>Logout</i>	-	a. Jika berhasil maka akan kembali ke tampilan menu <i>login</i> . b. Jika tidak berhasil maka akan tetap pada <i>form</i> menu utama	Valid

- Pengujian Aplikasi Web

Pengujian *login* yaitu untuk mengecek apakah fungsional dari *form login* telah berjalan dengan baik atau belum. Dari pengujian yang dilakukan didapatkan kesimpulan bahwa *form login* pada aplikasi *web* telah berfungsi dengan baik. Tabel 4 merupakan pengujian *login* untuk aplikasi *web*.

Tabel 4 Pengujian *Black Box* Untuk *Form Login* Aplikasi Web

Aktivitas	Input	Output	Status Pengujian
<i>Login Admin, Kesiswaan, Wali Kelas, Administrasi</i>	<i>Username</i> dan <i>password</i>	a. Jika berhasil maka masuk ke menu utama b. Jika <i>password</i> salah maka muncul peringatan <i>password</i> salah	Valid

Pengujian menambah data yaitu untuk mengecek apakah fungsional dari *form* menambah data telah berjalan dengan baik atau belum. Dari pengujian yang dilakukan didapatkan kesimpulan bahwa *form* menambah data pada aplikasi *web* telah berfungsi dengan baik. Tabel 5 merupakan pengujian menambah data untuk aplikasi *web*.

Tabel 5 Pengujian *Black Box* Untuk *Form Login* Menambah Data

Aktivitas	Input	Output	Status Pengujian
-----------	-------	--------	------------------

Admin menambah data operator, data siswa, data pelanggaran	Data yang akan di <i>input</i>		a. Jika berhasil maka data akan muncul pada tabel b. Jika gagal maka akan tetap pada <i>form</i> tambah data dan <i>form</i> menjadi kosong	<i>Valid</i>
Guru menambah data nilai, data absensi	Data yang akan di <i>input</i>		c. Jika berhasil maka data akan muncul pada tabel a. Jika gagal maka akan tetap pada <i>form</i> tambah data dan <i>form</i> menjadi kosong	<i>Valid</i>
Kesiswaan menambah data pelanggaran	Memasukkan jenis pelanggaran		a. Jika berhasil maka akan tampil tabel data pelanggaran dan akumulasi <i>point</i> akan muncul sesuai dengan pelanggaran yang telah dilakukan b. Jika gagal maka akan tetap pada <i>form</i> tambah data dan <i>form</i> menjadi kosong	<i>Valid</i>
Wali Kelas menambah data catatan	Memasukkan catatan		a. Jika berhasil maka akan tampil tabel data catatan b. Jika gagal maka akan tetap pada <i>form</i> tambah data dan <i>form</i> menjadi kosong	<i>Valid</i>
Administrasi menambah data keuangan	Memasukkan data keuangan dan batas waktu pembayaran		a. Jika berhasil maka akan tampil tabel data keuangan dan keterangan lunas/ belum lunas b. Jika gagal maka akan tetap pada <i>form</i> tambah data dan <i>form</i> menjadi kosong	<i>Valid</i>

Berdasarkan pengujian yang dilakukan didapatkan hasil bahwa aplikasi ini sudah berjalan secara fungsional dan mengeluarkan informasi sesuai dengan yang diharapkan

b. Pengujian Beta

Pengujian beta dilakukan dengan menyebar kuisisioner kepada pengguna aplikasi, yaitu orang tua siswa sebagai pengguna aplikasi. Pengujian ini digunakan untuk mengetahui sejauh mana aplikasi ini berfungsi dengan baik. Kuisisioner dibagi sejumlah 99 kepada orang tua atau wali murid. Dalam penafsiran data digunakan metode penafsiran data [17]. Tabel 6 menunjukkan penafsiran data menggunakan dua angka di belakang koma.

Tabel 6 Tabel Penafsiran Persentase

Jumlah Persentase	Keterangan
0,00%	Tidak ada
0,01% - 24,99%	Sebagian kecil
25% - 49,99%	Hampir setengah
50%	Setengahnya
50,01% - 74,99%	Sebagian besar
75% - 99,99%	Pada umumnya
100%	Seluruhnya

Dari hasil pengisian kuisisioner nomor 1 maka diperoleh hasil bahwa sebanyak 46 responden atau sebesar 46,46% menjawab “sangat mudah”. Berarti hampir setengah responden menyatakan bahwa aplikasi sangat mudah digunakan. 42 responden atau sebesar 42,42% menjawab “mudah”.

Berarti hampir setengah responden menyatakan bahwa aplikasi mudah digunakan. 9 responden atau sebesar 9,10% menjawab “cukup mudah”. Berarti sebagian kecil responden menyatakan bahwa aplikasi cukup mudah digunakan. Dan 2 responden atau sebesar 2,02% menjawab “sulit”. Berarti sebagian kecil responden menyatakan bahwa aplikasi sulit digunakan.

Dari hasil pengisian kuisioner nomor 2 maka diperoleh hasil bahwa sebanyak 81 responden atau sebesar 81,82% menjawab “sangat membantu”. Berarti pada umumnya responden menyatakan bahwa aplikasi sangat membantu dalam proses *monitoring* siswa. 13 responden atau sebesar 13,13% menjawab “membantu”. Berarti sebagian kecil responden menyatakan bahwa aplikasi membantu dalam proses *monitoring* siswa. 5 responden atau sebesar 5,05% menjawab “cukup membantu”. Berarti sebagian kecil responden menyatakan bahwa aplikasi cukup membantu dalam proses *monitoring* siswa. Dan tidak ada responden yang menyatakan bahwa aplikasi tidak membantu dalam proses *monitoring*.

Dari hasil pengisian kuisioner nomor 3 maka diperoleh hasil bahwa sebanyak 84 responden atau sebesar 84,85% menjawab “ya”. Berarti pada umumnya responden menyatakan bahwa aplikasi ini lebih fleksibel daripada sistem *monitoring* yang ada sebelumnya. 15 responden atau sebesar 15,15% menjawab “tidak”. Berarti sebagian kecil responden menyatakan bahwa aplikasi ini tidak lebih fleksibel daripada sistem *monitoring* yang ada sebelumnya.

Dari hasil pengisian kuisioner nomor 4 bahwa sebanyak 84 responden atau sebesar 84,85% menjawab “ya”. Berarti pada umumnya responden setuju jika sistem *monitoring* yang ada saat ini diganti dengan aplikasi ini. Dan sebanyak 15 responden atau sebesar 15,15% menjawab “tidak”. Berarti sebagian kecil responden tidak setuju jika sistem *monitoring* yang ada saat ini diganti dengan aplikasi ini.

5. Kesimpulan

Berdasarkan hasil penelitian dan pembahasan maka dapat diambil kesimpulan bahwa *student monitoring system* ini dapat diimplementasikan dengan baik dalam bentuk aplikasi J2ME pada perangkat *mobile* yang menyediakan informasi bagi orang tua mengenai nilai, absensi, pelanggaran dan data keuangan siswa. Library NuSOAP digunakan untuk mengirim dan menerima pesan SOAP. NuSOAP menyediakan berbagai fungsi *server* maupun *client* yang diimplementasikan dalam PHP. Untuk akses J2ME ke *web service* digunakan *library* JSR 172 yang mendefinisikan *web service* API yaitu fungsi-fungsi dasar yang digunakan dalam *web service client*.

6. Daftar Pustaka/Pustaka Acuan

- [1] <http://www.asiamediajournal.com/pressrelease.php?id=3756>. (diakses tanggal 24 Juli 2012)
- [2] Deni, Adam Nur. 2011. *Pengembangan Aplikasi Chat Bluetooth Pada Mobile Phone Dengan J2ME*. Universitas Mercu Buana.
- [3] Data SMK Telekomunikasi Tunas Harapan. 2011.
- [4] Arsip Kesiswaan. 2012.
- [5] Ramadhan, Tarmizi. 2008. *Hubungan Pelanggaran dan Prestasi Belajar Siswa di Sekolah*.
- [6] Putra, Moch Awang. 2011. *Sistem Monitoring Presensi dan Akademik Siswa Berbasis SMS Gateway (Studi Kasus: SMA Persatuan Tulangan)*. Politeknik Telkom Bandung.
- [7] Wellem, Theophilus. 2009. *Perancangan Prototype Aplikasi Mobile Untuk Pengaksesan Web Service*. Salatiga: UKSW.