

APLIKASI BERBASIS WEB PENGOLAHAN NILAI AKADEMIK (STUDI KASUS: SMP NEGERI 34 BANDUNG)

Ulfah Rahmawati¹, Bayu Rima Aditya², Wawa Wikusna³

¹Program Studi D3 Manajemen Informatika, ²Fakultas Ilmu Terapan, ³Universitas Telkom

¹ulfahra5@gmail.com, ²bayu@tass.telkomuniversity.ac.id,

³wawa_wikusna@tass.telkomuniversity.ac.id

Abstrak

SMP Negeri 34 Bandung merupakan salah satu sekolah yang menerapkan kurikulum 2013 sebagai sarana proses belajar mengajar. Di SMP Negeri 34 Bandung proses pengolahan dan penyimpanan data nilai siswa masih menggunakan *microsoft excel*, yang mempunyai kelemahan seperti data tidak terintegrasi, adanya kemungkinan duplikasi data, serta proses pencarian data yang membutuhkan waktu lama karena data disimpan dalam berbagai *file* dan *sheet* yang berbeda. Tujuan pembuatan aplikasi ini adalah membantu staf tata usaha dalam mengelola data siswa dan guru, membantu guru dalam input nilai siswa, membantu staf kurikulum dalam mengolah nilai siswa, serta terdapat fitur SMS Gateway untuk mengirimkan nilai UTS kepada orang tua siswa, sehingga orang tua tidak perlu menunggu rapor keluar untuk mengetahui nilai siswa. Serta adanya laporan nilai siswa kepada kepala sekolah untuk melihat perkembangan siswa. Aplikasi ini dibangun dengan metode *Waterfall*, dengan bahasa pemrograman PHP dan menggunakan *Framework Codeigniter* dan *Mysql* sebagai media penyimpanan data. SMS Gateway digunakan untuk memberikan informasi nilai kepada orang tua. Pengujian yang dilakukan pada aplikasi ini menggunakan *user acceptance testing*, *user usability testing*, dan *black box testing*. Dengan adanya aplikasi ini diharapkan akan lebih membantu kinerja guru-guru di SMP Negeri 34 Bandung.

Kata Kunci: Pengolahan Nilai, *Waterfall*, SMS Gateway, *MySql*, *PHP*.

Abstract

SMP Negeri 34 Bandung is one of the schools that apply 'Kurikulum 2013' as a means of the learning process. In SMP Negeri 34 Bandung, processing and data storage the students still use microsoft excel, who weaknesses as data at are not integrated, the possibility that duplicating data, and the search the data takes a long time that the data stored in various file and the different sheet. Generation purposes application of this is helping staff administrative in managing data students and teachers, help the teacher to input the students's score, help staff curriculum to processing students's score, and there are features sms gateway to send the middle test score to parents, so parents does not need to wait report out to know the students's score. And the headmaster can see the report students. Application is built with a waterfall method, with programming language php and use framework codeigniter and mysql as a medium data storage. Sms gateway used to inform the student's score to parents. For testing this application uses user acceptance testing, user usability testing, and black box testing. With the application is expected to be more helpful performance teachers in SMP Negeri 34 Bandung.

Keywords: Processing value, *Waterfall*, SMS Gateway, *MySql*, *PHP*.

1. Pendahuluan

1.1 Latar Belakang

Kurikulum adalah seperangkat rencana dan pengaturan mengenai tujuan, isi, dan bahan pelajaran serta cara yang digunakan sebagai pedoman penyelenggaraan kegiatan pembelajaran untuk mencapai tujuan pendidikan tertentu (UU nomor 20 tahun 2003; PP nomor 19 tahun 2005) [1]. Merujuk pada Peraturan Menteri Pendidikan dan Kebudayaan (Permendikbud) Nomor 58 Tahun 2014 Tentang Kurikulum 2013 Sekolah Menengah Pertama/Madrasah Tsanawiyah, Kurikulum yang telah dilaksanakan sejak tahun ajaran 2013/2104 disebut Kurikulum 2013 Sekolah Menengah Pertama/Madrasah Tsanawiyah [2].

Sesuai dengan acuan di atas, SMP Negeri 34 Bandung menggunakan kurikulum 2013 sebagai sarana proses

belajar mengajar. Salah satu hal yang berubah dalam Kurikulum 2013 dibanding dengan Kurikulum sebelumnya (Kurikulum 2006) adalah pola penilaian dalam rapor siswa yang tidak lagi menggunakan angka, melainkan dalam bentuk huruf atau deskriptif. Cakupan penilaian menurut Kurikulum 2013 dirumuskan menjadi empat bagian Kompetensi Inti (KI) yaitu, KI sikap spiritual, KI sikap sosial, KI pengetahuan, dan KI keterampilan.

Berdasarkan hasil wawancara dengan bagian kurikulum sebagai tim pengolah nilai diperoleh informasi bahwa pada penerapan kurikulum 2013, khususnya di SMP Negeri 34 Bandung terdapat kendala yaitu proses pengolahan dan penyimpanan data nilai siswa masih menggunakan *microsoft excel*, yang mempunyai kelemahan seperti data tidak terintegrasi, adanya kemungkinan duplikasi data dan redundansi data, dapat terjadi inkonsistensi data, serta

proses pencarian data yang membutuhkan waktu lama karena data disimpan dalam berbagai *file* dan *sheet* yang berbeda. Selain itu, kepala sekolah mengharapkan laporan nilai siswa tiap tahun ajaran, sehingga kepala sekolah dapat mengetahui perkembangan nilai siswa. Serta orang tua juga mengharapkan dapat mengetahui nilai siswa tanpa harus menunggu nilai rapor keluar.

Untuk mengatasi permasalahan tersebut, diusulkan untuk membangun suatu aplikasi berbasis *web* pengolahan nilai akademik di SMP Negeri 34 Bandung. Dengan adanya aplikasi ini diharapkan akan lebih memudahkan kerja guru dan juga staf kurikulum dalam pengolahan nilai serta kepala sekolah dan orang tua siswa dalam memberikan informasi.

1.2 Metode Pengerjaan

Dalam pengembangan aplikasi perlu digunakan metodologi *Software Development Life Cycle* (SDLC) sebagai pedoman bagaimana dan apa yang harus dilakukan selama melaksanakan pengembangan sistem. Adapun metode pengembangan yang digunakan adalah metode *Waterfall*. Secara garis besar model *Waterfall* mempunyai langkah-langkah seperti Analisis, Desain, Pengkodean, Pengujian [3].

1. Analisis

Analisis Kebutuhan kebutuhan perangkat lunak merupakan proses pengumpulan kebutuhan dilakukan secara intensif untuk memspezifikasikan kebutuhan perangkat lunak agar dapat dipahami seperti apa yang dibutuhkan oleh *user*. Pada tahapan ini dilakukan pengumpulan data terdiri dari data primer dan data sekunder, dimana sumber pengumpulan data primer dihasilkan dari:

- a. Observasi yaitu melakukan peninjauan langsung ke SMP Negeri 34 Bandung untuk mengetahui data siswa dan data guru.
- b. Wawancara dilakukan untuk pengumpulan semua kebutuhan *user* yang berkaitan dengan aplikasi yang akan dibangun, serta untuk mengetahui sistem yang sedang berjalan dan bagaimana keinginan dan permasalahan pengguna. Pada tahap ini, wawancara dilakukan kepada Guru dan Staf Kurikulum. Sedangkan sumber pengumpulan data sekunder dihasilkan dari:
 - a. Dokumen sekolah yaitu pengumpulan data yang dilakukan dengan mempelajari dokumen-dokumen yang ada untuk memperoleh data dan informasi penelitian.
 - b. Studi pustaka yaitu mempelajari masalah-masalah yang ada pada SMP Negeri 34 Bandung. Melakukan riset dengan merujuk pada buku-buku yang berhubungan dengan judul tugas akhir ini, guna untuk memecahkan masalah yang ada di SMP Negeri 34 Bandung.

Tahapan ini akan menghasilkan dokumen *user requirement* atau dapat dikatakan sebagai data yang berhubungan dengan keinginan *user* dalam pembuatan sistem. Hasil dari tahapan ini akan dijadikan dasar untuk ke tahapan selanjutnya.

2. Desain

Setelah mengumpulkan dan menganalisa data yang dibutuhkan, maka tahap selanjutnya yaitu menerjemahkan kebutuhan *user* menjadi desain teknis yang siap diimplementasikan seperti penyusunan proses, data, aliran proses dan hubungan antar data yang paling optimal untuk menjalankan proses bisnis dan memenuhi kebutuhan *user* sesuai dengan hasil analisis kebutuhan. Dokumentasi yang dihasilkan dari tahap ini antara lain *Flowmap*, *Use Case Diagram*, *Sequence Diagram*, *Class Diagram*, *Entity Relationship Diagram* hingga perancangan antarmuka untuk aplikasi yang akan dibangun. Hasil dari tahapan ini akan dijadikan dasar untuk melanjutkan ke tahap selanjutnya.

3. Pengodean

Pengodean merupakan penerjemahan desain dalam bahasa yang dapat dikenali oleh komputer. Pada tahap ini dilakukan *coding* yang sesuai dengan sistem yang telah di desain untuk membuat Aplikasi Pengolahan Nilai Akademik. Bahasa pemrograman yang digunakan yaitu PHP dengan *Framework Codeigniter* dan MySQL sebagai *database server*. Hasil dari tahapan ini adalah program komputer sesuai dengan desain yang telah dibuat pada tahap desain.

4. Pengujian

Setelah pengkodean selesai maka akan dilakukan pengujian terhadap sistem yang telah dibuat. Hal ini dilakukan untuk meminimalisir kesalahan (*error*) dan memastikan keluaran yang dihasilkan sesuai dengan keinginan. Pengujian yang digunakan menggunakan mengujian *black-box testing*, *user usability testing*, dan *user acceptance testing*.

2. Tinjauan Pustaka

2.1 Pengolahan

Pengolahan adalah proses, cara, perbuatan mengolah, data proses, cara, perbuatan mengolah data. Dalam hal ini data yang diolah adalah nilai [4].

2.2 Nilai

Nilai diartikan sebagai harga, dalam hal ini adalah suatu angka kepandaian [5].

2.3 Kurikulum 2013 (K-13)

Kurikulum ialah perangkat mata pelajaran yang diajarkan pada lembaga pendidikan [6]. Kurikulum 2013 (K-13) merupakan kurikulum tetap diterapkan oleh pemerintah untuk menggantikan Kurikulum Tingkat Satuan Pendidikan yang telah berlaku selama lebih 6 tahun. Kurikulum 2013 masuk dalam masa percobaannya tahun 2013 dengan menjadikan beberapa sekolah menjadi sekolah rintisan. Merujuk

pada Peraturan Menteri Pendidikan dan Kebudayaan (Permendikbud) Nomor 58 Tahun 2014 Tentang Kurikulum 2013 Sekolah Menengah Pertama/Madrasah Tsanawiyah, Kurikulum yang telah dilaksanakan sejak tahun ajaran 2013/2104 disebut Kurikulum 2013 Sekolah Menengah Pertama/Madrasah Tsanawiyah [2].

2.4 Konsep Penilaian Kurikulum 2013 di SMP

Penilaian setiap mata pelajaran meliputi kompetensi pengetahuan, kompetensi keterampilan, dan kompetensi sikap. Kompetensi pengetahuan dan kompetensi keterampilan menggunakan skala 1–4 (kelipatan 0.33), yang dapat dikonversi ke dalam Predikat A - D sedangkan kompetensi sikap menggunakan skala Sangat Baik (SB), Baik (B), Cukup (C), dan Kurang (K). Penilaian yang dilakukan untuk mengisi laporan hasil belajar ada tiga macam yaitu [7]:

1. Penilaian Kompetensi Pengetahuan
 - a. Penilaian Kompetensi Pengetahuan dilakukan oleh guru mata pelajaran (Pendidik).
 - b. Penilaian Pengetahuan terdiri atas:
 - 1) Nilai Ulangan Harian (UH)
 - 2) Nilai Ulangan Tengah Semester (UTS) / MID
 - 3) Nilai Ulangan Akhir Semester (UAS) / Ulangan Kenaikan Kelas (UKK)
 - c. Perhitungan Nilai Pengetahuan diperoleh dari rata-rata nilai Ulangan Harian (UH), Nilai Ulangan Tengah Semester (UTS) / MID, Nilai Ulangan Akhir Semester (UAS) / Ulangan Kenaikan Kelas (UKK).
 - d. Penentuan Nilai untuk Kompetensi Pengetahuan menggunakan rentang nilai.
 - e. Perhitungan Nilai Pengetahuan adalah dengan cara:
 - 1) Menggunakan skala nilai 0 sampai dengan 100.
 - 2) Rumus:

$$\begin{aligned} \text{Nilai Pengetahuan} &= \text{UH1} + \text{UH2} + \text{UH3} + \text{UTS} + \text{UAS} \\ \text{RT UH} &= (\text{UH1} + \text{UH2} + \text{UH3}) / 3 \\ \text{NR} &= (60\% * \text{RT UH}) + (20\% * \text{UTS}) + (20\% * \text{UAS}) \end{aligned}$$
 Keterangan :

UH	= Ulangan Harian
UTS	= Ulangan Tengah Semester
UAS	= Ulangan Akhir Sekolah
RT UH	= Rata-rata Ulangan Harian
NR	= Nilai Rapor
2. Penilaian Kompetensi Keterampilan
 - a. Penilaian Kompetensi Keterampilan dilakukan oleh Guru Mata Pelajaran (Pendidik).
 - b. Penilaian Kompetensi Keterampilan diperoleh melalui penilaian kinerja yang terdiri atas:
 - 1) Nilai Praktik
 - 2) Nilai Proyek

- 3) Nilai Portofolio
- c. Perhitungan Nilai Keterampilan adalah dengan cara:
 - 1) Menggunakan skala 0 sampai dengan 100.
 - 2) Rumus:

$$\begin{aligned} \text{Nilai Keterampilan} &= \text{Nilai Praktik} + \text{Nilai Proyek} + \text{Nilai Portofolio} \\ \text{Nilai Rapor} &= (\text{Nilai Praktik} + \text{Nilai Proyek} + \text{Nilai Portofolio}) / 3 \end{aligned}$$
- d. Penentuan Nilai untuk Kompetensi Keterampilan menggunakan rentang nilai.
3. Penilaian Kompetensi Sikap
 - a. Penilaian Kompetensi Sikap dilakukan oleh guru mata pelajaran (Pendidik).
 - b. Penilaian sikap diperoleh menggunakan:
 - 1) Penilaian observasi
 - 2) Penilaian diri sendiri
 - 3) Penilaian teman

Nilai observasi, diri, dan teman diperoleh dari hasil pengamatan terhadap proses sikap tertentu pada sepanjang proses pembelajaran satu Kompetensi Dasar (KD).

2.5 Model Waterfall

Model SDLC air terjun (*waterfall*) sering juga disebut model sekuensial linier (*sequential linear*) atau alur hidup klasik (*classic life cycle*). Model air terjun menyediakan pendekatan alur hidup perangkat lunak secara sekuensial atau terurut dimulai dari analisis, desain, pengkodean, pengujian, dan tahap pendukung (*support*).

2.6 Aplikasi Berbasis Web

Menurut [8] Aplikasi berbasis *web* merupakan “aplikasi yang dijalankan melalui *browser* dan tidak perlu diinstall dulu”. Keunggulan aplikasi berbasis *web* ini antara lain sebagai berikut:

- a. *Platform Independent*, artinya aplikasi ini dapat dijalankan dari sistem operasi windows, linux, BSD, Mac.
- b. Untuk dijalankan di banyak komputer, anda tidak perlu install di aplikasi disetiap komputer, cukup meng-*copy script* programnya ke *server* atau salah satu komputer. Untuk komputer lain yang ingin menjalankan program ini cukup buka *browser*-nya dan membuka alamat *host server* dimana program ini disimpan.
- c. Aplikasi ini dapat dijalankan dari jarak jauh dengan menggunakan internet. Aplikasi berbasis *web* ini dibuat dengan menggunakan program *server side*, contohnya ASP, JSP, PHP, dan bahasa lainnya. ASP (*Active Server Side*) merupakan turunan dari MS Visual Basic yang dibuat oleh Microsoft. JSP (*Java Server Page*)

merupakan bahasa berbasis Java yang dikembangkan oleh *Sun Microsystems* Sedangkan PHP (*PHP Hypertext PreProcessor*) merupakan bahasa pemrograman yang paling banyak komunitasnya saat ini dan dikembangkan secara *open source*.

2.7 Flowmap

Flowmap merupakan bagian alir yang menggambarkan arus dokumen - dokumen dan laporan - laporan termasuk tembusan - tembusannya pada sebuah program atau sistem secara logika. *Flowmap* mendefinisikan hubungan antara bagian (pelaku proses), proses manual atau berbasis *computer* dan aliran data (dalam bentuk dokumen keluaran dan masukan) [9].

2.8 Use Case Diagram

Use case atau diagram *use case* merupakan pemodelan untuk kelakuan (*behaviour*) sistem informasi yang akan dibuat. *Use case* mendeskripsikan sebuah interaksi antara satu atau lebih aktor dengan sistem informasi yang akan dibuat. Secara kasar, *use case* digunakan untuk mengetahui fungsi apa saja yang ada di dalam sebuah sistem informasi dan siapa saja yang berhak menggunakan fungsi-fungsi itu. [3]

2.9 Sequence Diagram

Sequence Diagram menggambarkan kelakuan objek pada *use case* dengan mendeskripsikan waktu hidup objek dan *message* yang dikirimkan dan diterima antar objek. Oleh karena itu untuk menggambarkan *sequence diagram* maka harus diketahui objek-objek yang terlibat dalam sebuah *use case* beserta metode-metode yang dimiliki kelas yang diinstansiasi menjadi objek itu. Banyak *sequence diagram* yang harus digambar adalah sebanyak pendefinisian *use case* yang memiliki proses sendiri atau yang paling penting semua *use case* yang telah didefinisikan interaksi jalannya pesan sudah dicakup pada *sequence diagram* sehingga semakin banyak *use case* yang didefinisikan maka *sequence diagram* yang harus dibuat juga semakin banyak. Berikut adalah simbol- simbol yang ada pada *sequence diagram* [3].

2.10 Entity Relationship Diagram (ERD)

Entity Relationship Diagram (ERD) dikembangkan berdasarkan teori himpunan dalam bidang matematika. ERD digunakan untuk pemodelan basis data relasional. Sehingga jika penyimpanan basis data menggunakan OODBMS maka perancangan basis data tidak perlu menggunakan ERD [3].

2.11 Class Diagram

Diagram kelas atau *class diagram* menggambarkan struktur sistem dari segi pendefinisian kelas-kelas yang akan dibuat untuk membangun sistem. Kelas memiliki apa yang disebut atribut dan metode atau operasi.

- Atribut merupakan variabel-variabel yang dimiliki oleh suatu kelas.
- Operasi atau metode adalah fungsi-fungsi yang dimiliki oleh suatu kelas.

Diagram kelas dibuat agar pembuat program atau *programmer* membuat kelas-kelas sesuai rancangan

di dalam diagram kelas agar antara dokumentasi perancangan dan perangkat lunak sinkron [3].

2.12 Hypertext Markup Language (HTML)

HTML kependekan dari *Hyper Text Markup Language*. Dokumen HTML adalah file teks murni yang dapat dibuat dengan editor teks sembarang. Dokumen ini dikenal sebagai *web page*. Dokumen HTML merupakan dokumen yang disajikan dalam *browser web surfer*. Dokumen ini umumnya berisi informasi ataupun *interface* aplikasi di dalam Internet [10].

2.13 Cascading Style Sheets (CSS)

Cascading Style Sheets (CSS) menggambarkan bagaimana sebuah naskah dipresentasikan pada layar (*browser*). CSS juga memisahkan presentasi dari isi naskah, yakni suatu perubahan yang dibuat dalam CSS tanpa mengubah naskah (XHTML) akan mengubah secara otomatis bentuk presentasi naskah pada layar [11].

2.14 PHP Hypertext Preprocessor (PHP)

PHP merupakan *script* untuk pemrograman *script web server-side*, *script* yang membuat dokumen HTML secara *on the fly*, dokumen HTML yang dihasilkan dari suatu aplikasi buka dokumen HTML yang dibuat dengan menggunakan *editor* teks atau *editor* HTML. Dengan menggunakan PHP maka maintenancesuatu situs *web* menjadi lebih mudah. Proses *update* data dapat dilakukan dengan menggunakan aplikasi yang dibuat dengan menggunakan *script* PHP [12].

PHP (*Hypertext Preprocessor*) ini merupakan bahasa yang hanya dapat berjalan pada *server* yang hasilnya dapat ditampilkan pada klien. *Interpreter* PHP dalam mengeksekusi kode PHP pada sisi *server* (disebut *server-side*) berbeda dengan mesin maya *Java* yang mengeksekusi program pada sisi klien (*client-side*). PHP merupakan bahasa standar yang digunakan dalam dunia *website*. PHP adalah bahasa program yang berbentuk *script* yang diletakkan didalam *server web* [13].

2.15 Framework CodeIgniter

CodeIgniter adalah sebuah *framework* PHP yang dapat membantu mempercepat *developer* dalam pengembangan aplikasi berbasis PHP dibandingkan jika menulis semua kode program dari awal. *Framework* merupakan kumpulan potongan-potongan program yang disusun atau diorganisasikan sedemikian rupa, sehingga dapat digunakan untuk membantu membuat aplikasi utuh tanpa harus membuat kodenya dari awal [14].

2.16 Model View Controller (MVC)

Model View Controller (MVC) merupakan salah satu contoh dari *Architectural Design Pattern*. Konsep MVC bertujuan agar sebuah aplikasi dapat mudah dipelihara oleh orang-orang di dalam tim pengembangan yang berbeda spesifikasi pekerjaan,

misalnya *Database Administrator* (DBA) untuk mengurus masalah basis data, blok *controller* untuk *programmer*, dan blok *view* untuk desainer antarmuka (*interface designer*) [3].

2.17 My Structured Query Language (MySQL)

MySQL adalah sebuah program *database server* yang mampu menerima dan mengirimkan datanya dengan sangat cepat, *multi-user* serta menggunakan perintah standar SQL (*Structured Query Language*). MySQL adalah sebuah *server*, dapat juga berperan sebagai *client* sehingga sering disebut *database/server*, yang *open source* dengan kemampuan dapat berjalan baik di OS (*Operating Sistem*) manapun, dengan *Platform* Windows maupun Linux [15].

2.18 SMS Gateway

SMS Gateway adalah suatu *platform* yang menyediakan mekanisme untuk mengirim dan menerima SMS. *SMS Gateway* dapat berkomunikasi dengan perangkat lain yang memiliki *SMS platform* untuk menghantar dan menerima pesan SMS dengan sangat mudah [16].

2.19 Pengujian Black-Box

Pengujian *Black-Box* berfokus pada persyaratan fungsional perangkat lunak. Pengujian *Black-Box* memungkinkan perekayasa perangkat lunak mendapatkan serangkaian kondisi *input* yang sepenuhnya menggunakan semua persyaratan fungsional untuk suatu program. Pengujian *Black-Box* berusaha menemukan kesalahan dalam kategori sebagai berikut [17]:

1. Fungsi-fungsi yang tidak benar atau hilang.
2. Kesalahan *interface*.
3. Kesalahan dalam struktur data atau akses *database* eksternal.
4. Kesalahan kinerja.
5. Inisialisasi dan kesalahan terminasi.

2.20 Pengujian User Acceptance Testing

Menurut Perry (2006:70), *User Acceptance Testing* merupakan pengujian yang dilakukan oleh *end-user* dimana *user* tersebut adalah *staff/karyawan* perusahaan yang langsung berinteraksi dengan sistem dan dilakukan verifikasi apakah fungsi yang ada telah berjalan sesuai dengan kebutuhan/fungsinya [18].

Menurut Lewis (2009:134), setelah dilakukan *system testing*, *acceptance testing* menyatakan bahwa sistem *software* memenuhi persyaratan. *Acceptance testing* merupakan pengujian yang dilakukan oleh pengguna yang menggunakan teknik pengujian *black box* untuk menguji sistem terhadap spesifikasinya. Pengguna akhir bertanggung jawab untuk memastikan semua fungsionalitas yang relevan telah diuji [18].

2.21 Pengujian Usability

Usability adalah analisa kualitatif yang menentukan seberapa mudah *user* menggunakan antarmuka suatu aplikasi (Nielsen, 2012). Suatu aplikasi disebut *usable* jika fungsi-fungsinya dapat dijalankan secara

efektif, efisien, dan memuaskan (Nielsen, 1993). Pengujian *usability* dapat dilakukan dengan melibatkan pengguna atau tanpa melibatkan pengguna. Pengujian dengan melibatkan pengguna dapat memberikan informasi langsung dari pengguna tentang bagaimana pengguna menggunakan sistem serta permasalahan yang dihadapi. Pengujian ini terdiri atas metode *Field Observation* (observasi langsung), *Questionnaire* (kuesioner) dan *Thinking Aloud* (Holzinger 2005). Pada metode observasi langsung, seorang observator mengamati satu atau lebih pengguna di lokasi. Metode kuesioner dilakukan dengan menggunakan kuesioner untuk mengukur kepuasan pengguna dan untuk mengetahui opini pengguna terhadap aplikasi yang digunakan. Metode *Thinking Aloud* dilakukan dengan mengamati secara terus menerus sikap tubuh dan ucapan yang ditunjukkan oleh seorang responden pada saat menggunakan aplikasi [19].

3. Analisis dan Perancangan

3.1 Gambaran Sistem Saat Ini

Pada proses bisnis yang sedang berjalan di SMP Negeri 34 Bandung terdapat proses bisnis pengolahan nilai terpusat, artinya semua komponen nilai siswa dan presensi siswa direkap dan diolah disatu pihak yaitu bagian kurikulum. Proses input nilai dilakukan oleh masing-masing guru mata pelajaran dan dikumpulkan disatu pihak, yaitu bagian kurikulum untuk diolah dan direkap.

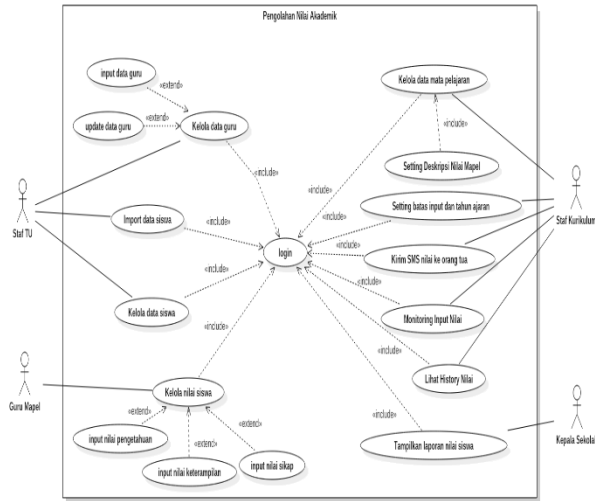
3.2 Analisis Sistem Usulan

Pada tahap sistem yang diusulkan menguraikan tentang aplikasi yang akan digunakan oleh guru mata pelajaran, staf kurikulum, staf tata usaha, dan kepala sekolah di SMP Negeri 34 Bandung. Dalam proses bisnis ini menjelaskan alur yang terjadi di dalam aplikasi berbasis web pengolahan nilai akademik.

Asumsi seluruh data-data yang dibutuhkan termasuk data guru dan siswa telah di-import ke dalam aplikasi. Setelah itu user mendapatkan username dan password sebagai akun untuk mengakses aplikasi tersebut.

3.3 Analisis Kebutuhan Sistem

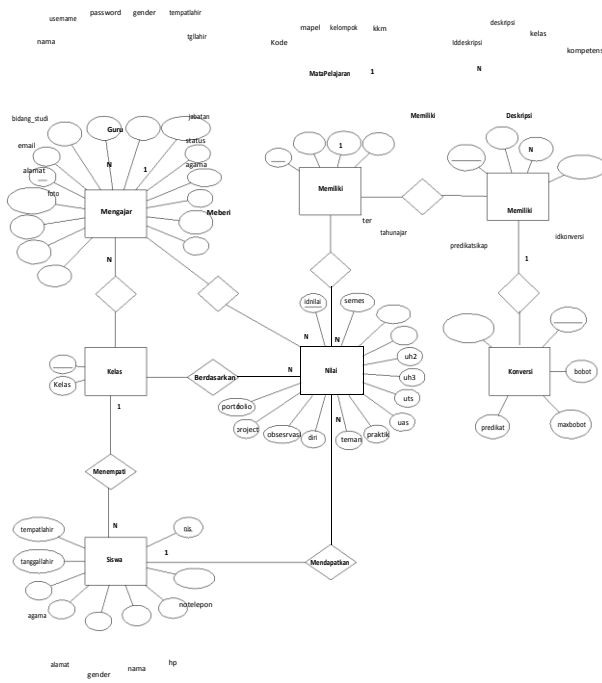
3.3.1 Perancangan Use Case Diagram



Gambar 3-1 Use Case Pengolahan Nilai Akademik

3.4 Perancangan Basis Data

3.4.1 Entity Relationship Diagram



Gambar 3-2 Entity Relationship Diagram Pengolahan Nilai Akademik

4. Implementasi dan Pengujian

4.1 Implementasi

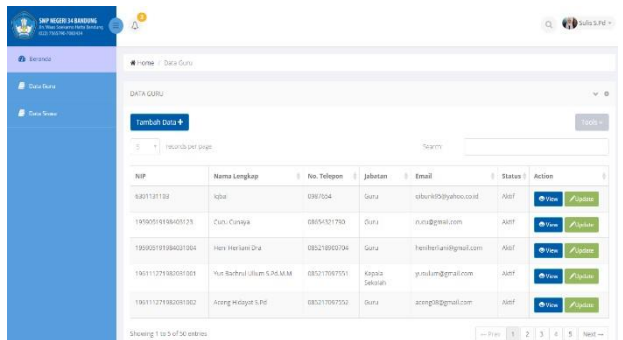
Implementasi merupakan tampilan aplikasi yang dibuat agar pengguna dapat berinteraksi dengan aplikasi yang dibangun. Berikut ini adalah antarmuka yang terdapat pada aplikasi ini.

4.1.1 Halaman Login



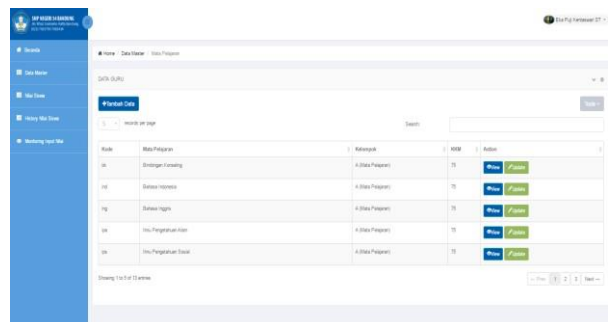
Gambar 4.1 Halaman Login

4.1.2 Halaman User Staf Tata Usaha



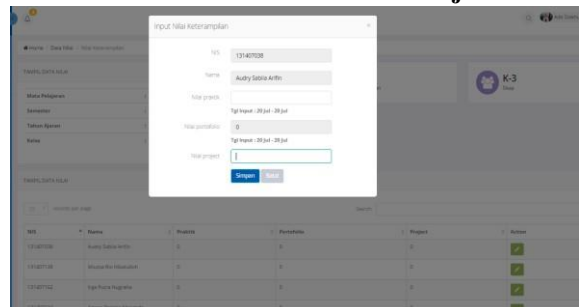
Gambar 4-1 Halaman Kelola Data Guru

4.1.3 Halaman User Staf Kurikulum



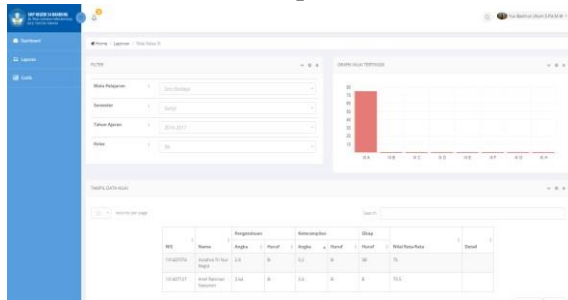
Gambar 4-2 Halaman Kelola Mata Pelajaran

4.1.4 Halaman User Guru Mata Pelajaran



Gambar 4.19 Halaman Input Nilai Keterampilan

4.1.5 Halaman User Kepala Sekolah



Gambar 4.5
Halaman Laporan Nilai Siswa

4.2 Pengujian

Pada tahap pengujian dijelaskan tentang proses pengujian yang dilakukan terhadap sistem yang dibangun. Pengujian dilakukan dengan *blackbox testing*, *user usability testing*, dan *user acceptance testing*.

5. Penutup

5.1 Kesimpulan

Berdasarkan tujuan dari pembangunan Aplikasi Berbasis Web Pengolahan Nilai Akademik, maka dapat ditarik kesimpulan sebagai berikut :

1. Aplikasi yang telah dibangun mampu memberikan fasilitas kepada guru mata pelajaran untuk melakukan input nilai siswa.
2. Dengan fitur pengolahan nilai, mampu memfasilitasi staf kurikulum dalam proses pengolahan nilai akademik.
3. Mampu memberikan laporan atau informasi mengenai nilai siswa kepada kepala sekolah.
4. Mampu memberikan informasi kepada orang tua siswa mengenai nilai siswa melalui *SMS Gateway*.

5.2 Saran

Saran yang dapat dipertimbangkan untuk pengembangan Aplikasi Berbasis Web Pengolahan Nilai Akademik (Studi Kasus: SMP Negeri 34 Bandung) ini di masa mendatang adalah :

1. Aplikasi ini dapat dikembangkan ke dalam aplikasi berbasis *android*.

6. Daftar Pustaka

- [1] Kemenag, "UU No. 20 Tahun 2003," [Online]. Available: <http://kemenag.go.id/file/dokumen/UU2003.pdf>. [Accessed Januari 2016].
- [2] kemdikbud, "SNP," [Online]. Available: <http://sdm.data.kemdikbud.go.id/SNP/snp.php>. [Accessed Januari 2016].
- [3] M. S. Rosa A.S, *Rekayasa Perangkat Lunak Terstruktur dan Berorientasi Objek*, Bandung: Informatika, 2015.

[4] K. B. B. I. (KBBI), "Kamus Besar Bahasa Indonesia," [Online]. Available: <http://kbbi.web.id/olah-2>. [Accessed Januari 2016].

[5] K. B. B. I. (KBBI), "Kamus Besar Bahasa Indonesia (KBBI)," [Online]. Available: <http://kbbi.web.id/nilai>. [Accessed Januari 2016].

[6] K. B. B. I. (KBBI), "Kamus Besar Bahasa Indonesia (KBBI)," [Online]. Available: <http://kbbi.web.id/kurikulum>. [Accessed Januari 2016].

[7] E. Kentaswari, Interviewee, Konsep Penilaian Kurikulum 2013. [Interview]. Januari 2016.

[8] Aswandi, *Aplikasi Berbasis Web*, Cirebon: Elex Media Komputindo, 2006.

[9] H. Jogiyanto, *Analisis dan Desain Sistem Informasi: Pendekatan Terstruktur Teori dan Praktik Aplikasi Bisnis*, Yogyakarta: Andi Offset, 2005.

[10] I. Betha Sidik and I. M. E. I. P. Husni, *Pemrograman Web dengan HTML*, Bandung: Informatika, 2005.

[11] A. Suhendar, *Teknologi Pemrograman Mobile Commerce*, Bandung: Informatika, 2003.

[12] I. Betha Sidik and I. M. E. I. P. Husni, *Pemrograman Web dengan PHP*, Bandung: Informatika, 2006.

[13] B. Nugroho, *PHP dan Mysql dengan Editor Dreamweaver MX*, Yogyakarta: Andi, 2004.

[14] A. Basuki, *Membangun Web Berbasis PHP dengan Framework Condeigniter*, Yogyakarta: Lokomedia, 2010.

[15] B. Nugroho, *Database Relasional dengan MySQL*, Yogyakarta: Andi, 2005.

[16] K. Wahana, *Mudah Membuat Aplikasi SMS Gateway dengan Codeigniter*, Jakarta: Elex Media Komputindo, 2014.

[17] R. S. Pressman, *Rekayasa Perangkat Lunak Pendekatan Praktisi (Buku Satu)*, Yogyakarta: Andi, 2002.

[18] L. Binus, "Bab 2 Landasan Teori," [Online]. Available: <http://library.binus.ac.id/eColls/eThesisdoc/Bab2DOC/2012-1-00007-SI%20Bab2001.doc>. [Accessed Juli 2016].

[19] Nurhadryani, "Pengujian Usability untuk Meningkatkan Antarmuka Aplikasi," 2013. [Online]. Available: <http://journal.ipb.ac.id/index.php/jika/article/view/7997/6284>. [Accessed Juli 2016].