**RANCANG BANGUN SISTEM MONITORING ACCESS POINT DENGAN MENGGUNAKAN METODE WATERFALL**

1Stevanya Alessandro Hamonangan

2Rini Agustina

1Sistem Informasi, Universitas Kanjuruhan Malang, sandroalpha@unikama.ac.id

2Sistem Informasi, Universitas Kanjuruhan Malang, riniagustina@unikama.ac.id

**Abstrak**

 Kegiatan monitoring jaringan diperlukan untuk mempermudah administrator dalam mengelola jaringan. Teknologi nirkabel atau sering disebut *wireless* sudah banyak diimplementasikan di berbagai tempat. Efisiensi teknologi nirkabel membuat banyak digunakan di berbagai sektor jaringan. Penilitian ini dirancang aplikasi web monitoring dengan menampilkan informasi mengenai perangkat *Access Point*. Penelitian ini menggunakan metode *waterfall*. Aplikasi yang dihasilkan mampu menampilkan informasi perangkat jaringan, seperti kondisi power, download, upload dan lokasi perangkat jaringan. Hasil uji coba menunjukan bahwa aplikasi dapat menampilkan informasi-infromasi yang dimonitor dengan baik. Hal ini ditunjukan dengan berjalannya sistem monitoring

**Kata Kunci:** *Access Point, Monitoring, Waterfall*

1. **Pendahuluan**

Salah satu teknologi nirkabel yang paling sering digunakan adalah titik akses nirkabel atau lebih sering disebut *Access Point Wireless*. *Access Point* adalah suatu perangkat yang dapat menciptakan sistem jaringan lokal secara nirkabel. Jaringan ini sering diimplementasikan pada penyebaran jaringan dalam skala besar, sehingga dengan hanya menggunakan satu perangkat saja telah dapat menghubungkan beberapa perangkat dalam satu waktu. Efisiensi ini yang membuat *Access Point* banyak digunakan di berbagai sektor jaringan.

Dalam beberapa situasi terkadang pengguna mengalami gangguan dalam mengakses jaringan nirkabel. Permasalahan tersebut bisa berupa terputusnya koneksi nirkabel atau menurunnya kualitas transmisi data nirkabel. Salah satu penyebab terjadinya permasalahan ini karena adanya penurunan performa perangkat *Access Point*. Hal ini dapat menyebabkan ketidakpuasan akan kualitas *wifi* yang diterima oleh pengguna.

Solusi yang bisa dilakukan oleh *administrator* jaringan adalah dengan melakukan pemeriksaan perangkat *Access Point* secara berkala untuk mengetahui kualitas dan performa perangkat tersebut. Hal ini bertujuan untuk mendapatkan informasi mengenai kondisi dari perangkat *Access Point* sehingga tindakan preventif dapat segera dilakukan seperti melakukan peremajaan pada perangkat yang telah mengalami penurunan kualitas dan performa. Tetapi indikator kualitas dari perangkat *Access Point* seringkali hanya berdasarkan asumsi atau pencatatan secara manaual dan tindakan baru bisa diambil jika terjadi komplain. Adanya sistem monitoring yang terintegrasi pada setiap perangkat akan sangat membantu *administrator* dalam menghindari kejadian seperti itu. Sistem monitoring secara sederhana akan bertindak sebagai penyaji informasi berdasarkan indikator tertentu, dalam hal ini sistem akan memeriksa kondisi perangkat nirkabel dalam interval tertentu untuk memberikan gambaran mengenai kondisi dari perangkat nirkabel yang berada dalam jaringan yang sedang dikelola.

Dalam penelitian ini akan menggunakan metode air terjun atau *waterfall*. Secara sederhana, metode ini merupakan metode yang menggunakan bentuk sistematis dan terstruktur pada pembuatan dan pengembangan perangkat lunak yang dimulai dari meneliti kebutuhan pengguna hingga diakhiri dengan penyerahan hingga memberikan pelayanan dukungan terhadap produk tersebut (Pressman, 2012). Melihat dari sistematis proses tersebut, maka metode ini sangat tepat digunakan dalam pengembangan suatu produk sistem informasi.

1. **Metode Penelitian**



**Gambar 1 Tahapan Metode Waterfall**

Adapun langkah-langkah yang dilakukan oleh penulis dengan dalam melakukan pengembangan sistem dibangun dengan mengimplementasikan model pengembangan waterfall dengan tahapan waterfall yaitu: 1) Analisis Kebutuhan, 2) Desain Sistem, 3) Implementasi Sistem, 4) Operasi dan Pemeliharaan.

Pada tahap analisis dilakukan analisis kebutuhan secara intensif untuk menspesifikasikan kebutuhan perangkat lunak agar dapat dipahami perangkat lunak seperti apa yang dibutuhkan user. Pada tahapan desain sistem dilakukan proses pembuatan struktur data, arsiterktur, antarmuka, dan prosedur pengkodean. Pada tahap implementasi sistem dilakukan proses pemrograman aplikasi. Pada tahap operasi dan pemeliharaan dilakukan pengujian fungsi pada sistem dan untuk pemeliharaan dilakukan secara berkala sesuai dengan kebutuhan mengulangi proses pengembangan mulai dari analisis spesifikasi untuk membuat perubahan perangkat lunak yang sudah ada, tapi tidak untuk membuat perangkat lunak baru. Dengan menggunakan model pengembangan waterfall sistem yang akan dibuat akan lebih terstruktur, dan mudah dipahami terutama dalam proses pembuatan sistem. Pada penelitian ini metode *waterfall* melingkupi aktivitas-aktivitas seperti:

1. **Analisis kebutuhan**

Dalam analisis kebutuhan merupakan tahapan dimana peneliti menganilis dan memahami sistem yang ada, mengidentifikasi masalah dan mencari solusi serta menentukan kebutuhan sistem. Apabila terjadi perubahan kebutuhan sistem maka proses kembali kepada awal menganalisi kelayakan sisten. Sehingga proses berikutnya dapat dilakukan apabila proses pertama telah selesai.

1. **Desain Sistem**

Desain merupakan tahap perancangan antarmuka, database, dan proses sistem. Berdasarkan analisis kebutuhan sistem maka dapat diketahui apa saja yang menjadi kebutuhan dari pengembangan sistem, sehingga sistem yang dibuat nantinya sesuai dengan apa yang diharapkan. Pada tahap desain sisten, peneliti melanjutkan dari tahapan analisis kebutuhan. Tahapan desain sistem dalam penelitian ini dilakukan dengan 2 tahapan yaitu :

1. Pemodelan perangkat lunak dengan menggunakan UML (Unified Modelling Language) yang akan dibagi menjadi Use case, Sub use case, Activity diagram, Sequence diagram dan Class diagram sebagai perancangan sistem.
2. Desain tampilan, pada tahapan ini peneliti akan melakukan desain tampilan *(user interface).*
3. **Implementasi Sistem**

Berdasarkan tahapan - tahapan desain sistem yang telah direncakan, peneliti akan melakukan proses penterjemahan desain sistem ke dalam bahasa pemrograman dalam hal ini menggunakan bahasa pemrograman HTML dan PHP serta database MySQL. Perangkat lunak pendukung yang digunakan yakni Macromedia Dreamwaver 8, Xampp, serta software lain yang mendukung dalam pembuatan program.

Setelah melalui proses pemrograman maka selanjutnya sistem yang dibangun akan dilakukan pengujian untuk mengetahui layak atau tidaknya untuk diimplementasikan.

1. **Operasi dan Pemeliharaan**

Tahap akhir model *waterfall*. Perangkat lunak yang sudah jadi, dijalankan serta dilakukan pemeliharaan. Pemeliharaan termasuk memperbaiki kesalahan yang

tidak ditemukan pada langkah sebelumnya

Pada penelitian ini, dilakukan perancangan sistem perangkat lunak untuk menggambakan prosedur dan proses kerja dari sistem aplikasi. Sistem dibangun dengan menggunakan UML (*Unified Modeling Language)* yang terdiri dari *use case diagram*, *activity diagram* dan *sequence diagram*. *Use Case Diagram* dapat dilihat pada gambar 2.



**Gambar 2 *Use Case Diagram***

*Class Diagram* merupakan hubungan antar kelas dan penjelasan detail tiap-tiap kelas di dalam model desain dari suatu sistem, juga memperlihatkan aturan-aturan dan tanggung jawab entitas yang menentukan perilaku sistem. *Class Diagram* juga menunjukkan atribut-atribut dan operasi-operasi dari sebuah kelas dan *constraint* yang berhubungan dengan objek yang dikoneksikan. *Class Diagram* menjadi kekuatan dasar dari metode-metode yang saling berinteraksi yang dapat dilihat pada gambar 3.

**Gambar 3 *Class Diagram***

1. **Hasil dan Pembahasan**
2. **Login**

Tampilan halaman *login* digambarkan seperti gambar 4halaman *login* dibawah ini.



**Gambar 4 Halaman Login**

**Tabel 1 Pengujian Sistem (Blackbox Testing) Login**

|  |
| --- |
| **Kode Testing :** PSBT2 |
| **Deskripsi :** *user* melakukan *login* untuk masuk ke halaman utama, *login* sesuai dengan hak akses kemudian memasukan *username* dan *password*, kemudian sistem merespon dengan melakukan pengecekan pada *database* |
| **No** | **Fungsi yang Diuji** | **Skenario****Pengujian** | **Hasil yang****Diharapkan** | **Hasil****Pengujian** | **Keterangan** |
| 1 | Login sebagai administrator | Masukan *username* dan *password* sesuai dengan hak akses | Sistem berhasil *login* | Sistem berhasil *login* sebagai administrasi | Sesuai |
| 2 | Login sebagai kepala | Masukan *username* dan *password* sesuai dengan hak akses | Sistem berhasil *login* | Sistem berhasil *login* sebagai kepala | Sesuai |
| 3 | Login sebagai teknisi | Masukan *username* dan *password* sesuai dengan hak akses | Sistem berhasil *login* | Sistem berhasil *login* sebagai teknisi | Sesuai |

Di atas merupakan pengujian sistem (*blackbox testing*) halaman login tahap 1 yang dapat dilihat pada tabel 1 pengujian sistem (*blackbox testing*) halaman login.

1. **Master Data User**

Tampilan halaman master data user digambarkan seperti gambar 5 halaman master data user dibawah ini.



**Gambar 5 Master Data User**

Berikut merupakan pengujian sistem (*blackbox testing*) halaman master data user tahap 1 yang dapat dilihat pada tabel 2 sebagai berikut :

**Tabel 2 Pengujian Sistem (Blackbox Testing) Master Data User**

|  |
| --- |
| **Kode Testing :** PSBT3 |
| **Deskripsi :** administrator melakukan tambah data user sistem, lalu menyimpan data user, lalu edit dan hapus data jabatan kemudian sistem merespon dengan update data pada *database* |
| **No** | **Fungsi yang Diuji** | **Skenario****Pengujian** | **Hasil yang****Diharapkan** | **Hasil****Pengujian** | **Keterangan** |
| 1 | Tambah Data User | Tambah data user | Data masuk di database | Data berhasil tersimpan di database  | Sesuai |
| 2 | Edit Data User | Edit data user | Data user dapat diubah dan pada database teredit | Data user pada database berhasil teredit  | Sesuai |
| 3 | Hapus Data User | Hapus data user | Data user pada databse terhapus | Data user pada database terhapus | Sesuai |

**3. Master Data Jabatan**

Tampilan halaman master data jabatan seperti gambar 6 :



**Gambar 6 Halaman Master Data Jabatan**

**Tabel 3 Pengujian Sistem (Blackbox Testing) Master Data Jabatan**

|  |
| --- |
| **Kode Testing :** PSBT4 |
| **Deskripsi :** administrator melakukan tambah data jabatan untuk user, lalu edit dan hapus data jabatan kemudian sistem merespon dengan update data pada *database* |
| **No** | **Fungsi yang Diuji** | **Skenario****Pengujian** | **Hasil yang****Diharapkan** | **Hasil****Pengujian** | **Keterangan** |
| 1 | Tambah Data Jabatan | Tambah data jabatan | Data masuk di database | Data berhasil tersimpan di database  | Sesuai |
| 2 | Edit Data Jabatan | Edit data jabatan | Data jabatan dapat diubah dan pada database teredit | Data jabatan pada database berhasil teredit | Sesuai |
| 3 | Hapus Data Jabatan | Hapus data jabatan | Data jabatan pada databse terhapus | Data jabatan pada database terhapus | Sesuai |

1. **Master Data AP**

Tampilan halaman master data ap digambarkan seperti gambar 7 :



**Gambar 7 Halaman Master Data AP**

**Tabel 4 Pengujian Sistem (Blackbox Testing) Master Data AP**

|  |
| --- |
| **Kode Testing :** PSBT5 |
| **Deskripsi :** administrator melakukan tambah data ap, lalu edit dan hapus data ap kemudian sistem merespon dengan update data pada *database* |
| **No** | **Fungsi yang Diuji** | **Skenario****Pengujian** | **Hasil yang****Diharapkan** | **Hasil****Pengujian** | **Keterangan** |
| 1 | Tambah Data AP | Tambah data ap | Data masuk di database | Data berhasil tersimpan di database | Sesuai |
| 2 | Edit Data AP | Edit data ap | Data ap dapat diubah dan pada database teredit | Data ap pada database berhasil teredit | Sesuai |
| 3 | Hapus Data AP | Hapus data ap | Data ap pada databse terhapus | Data ap pada database terhapus | Sesuai |

1. **Hak Akses AP**



**Gambar 8 Halaman Hak Akses AP**

Berikut merupakan pengujian sistem (*blackbox testing*) halaman master usehak akses ap tahap 1 yang dapat dilihat pada tabel 5 sebagai berikut :

**Tabel 5 Pengujian Sistem (Blackbox Testing) Hak Akses AP**

|  |
| --- |
| **Kode Testing :** PSBT6 |
| **Deskripsi :** administrator memilih menu hak akses ap, kemudian administrator dapat melakukan edit, kemudian sistem dapat melakukan perubahan pada *database* |
| **No** | **Fungsi yang Diuji** | **Skenario****Pengujian** | **Hasil yang****Diharapkan** | **Hasil****Pengujian** | **Keterangan** |
| 1 | Edit Hak Akses AP | Edit data hak akses ap | Data hak akses ap dapat diubah dan pada database teredit | Data hak akses ap pada database berhasil teredit | Sesuai |

1. **Laporan Per Hari**



**Gambar 9 Halaman Laporan Per Hari**

Berikut merupakan pengujian sistem (*blackbox testing*) halaman laporan Per Hari tahap 1 yang dapat dilihat pada tabel 6 pengujian sistem (*blackbox testing*) halaman laporan per hari sebagai berikut :

**Tabel 6 Pengujian Sistem (Blackbox Testing) Laporan Per Hari**

|  |
| --- |
| **Kode Testing :** PSBT7 |
| **Deskripsi :** kepala dan administrator memilih menu laporan per hari, kemudian sistem dapat menampilkan cetak laporan keuangan |
| **No** | **Fungsi yang Diuji** | **Skenario****Pengujian** | **Hasil yang****Diharapkan** | **Hasil****Pengujian** | **Keterangan** |
| 1 | Laporan Per Hari | Tampil laporan per hari | Sistem tampil laporan per hari | Sistem berhasil tampil laporan per hari  | Sesuai |

1. **Laporan Per Bulan**



**Gambar 10 Halaman Laporan Per Bulan**

Berikut merupakan pengujian sistem (*blackbox testing*) halaman laporan Per Bulan tahap 1 yang dapat dilihat pada tabel 7 pengujian sistem (*blackbox testing*) halaman laporan per bulan sebagai berikut :

**Tabel 7 Pengujian Sistem (Blackbox Testing) Laporan Per Bulan**

|  |
| --- |
| **Kode Testing :** PSBT7 |
| **Deskripsi :** kepala dan administrator memilih menu laporan per bulan, kemudian kepala dan administrator dapat melakukan cetak laporan per bulan kemudian sistem dapat menampilkan cetak laporan per bulan |
| **No** | **Fungsi yang Diuji** | **Skenario****Pengujian** | **Hasil yang****Diharapkan** | **Hasil****Pengujian** | **Keterangan** |
| 1 | Laporan Per Bulan | Pilih cetak kemudian akan tampil cetak laporan per bulan | Sistem tampil laporan per bulan | Sistem berhasil tampil cetak laporan per bulan | Sesuai |

**4. Kesimpulan**

Berdasarkan hasil penelitian rancang bangun sistem informas*i* monitoring *access point* dan inventaris dengan menggunakan metode waterfall:

1. Sistem membantu administrator jaringan untuk memperoleh informasi saat mengelola *access point*.
2. Peran UML dalam membangun sistem monitoring *access point* di unit MSI Universitas Kanjuruhan Malang dapat sesuai dengan kebutuhan pengguna yang terlibat yaitu (Administrator, Kanit Jaringan dan Teknisi).

**5. Saran**

Adapun saran untuk peneliti selanjutnya yaitu :

1. Sistem ini nanti diharapkan memiliki notifikasi dini untuk perangkat *access point* yang bermasalah.
2. Sistem ini diharapkan untuk penelitian berikutnya dapat dikembangkan dengan sistem *Simple Network Management Protocol*.

**Referensi**

Hendini, 2016. “Pemodelan UML Manajemen Sistem Informasi Monitoring Penjualan Dan Stok Barang (Studi Kasus Distro Zhezha Pontianak)”, *Manajemen Informatika*, STMIK Amik Pontianak, Desember 2016.

Dennis, Alan, Barbara Halley Wixom, Roberta M.Roth. 2012. “System Analysis and Desain”, Fifth Edition. New Jersey: John wiley & Sons, ISBN 978-1-118-05762-9.

Ermatita. 2016, “Analisis Dan Perancangan Sistem Informasi Perpustakaan”, *Jurnal Sistem Informasi (JSI),* Sriwijaya : Universitas Sriwijaya, ISSN Online : 2355-4614, VOL. 8, NO. 1, April 2016.

Lukas, Tri., Christanto, 2016, “Perancangan Sistem Monitoring Jaringan Melalui Perangkat Android Berbasis Simple Network Management Protocol”, Salatiga : Universitas Kristen Satya Wacana JANUARI 2016.

Loveri, Tomi. 2018, “Sistem Informasi Aplikasi Pengelolaan Transaksi Keuangan Dan Pendataan Konsumen Pada CV. Puplas”, *Jurnal Sains dan Informatika,* Padang : STMIK Jaya Nusa, E-ISSN : 2502-096X, V4.I2 (138-149) 2018.

Kadir, A., & Triwahyuni, T. C(2003), “Pengenalan Teknologi Informasi” : Yogyakarta.

Prayitno, Agus. 2015. “Pemanfaatan Sistem Informasi Perpustakaan Digital Berbasis Website Untuk Para Penulis”, , *Indonesian Journal on Software Engineering,* Jakarta Timur : AMIKBSI Jakarta Timur, ISSN : 2461‐0690, Volume 1 No 1 – 2015.

Satzinger, Jhon w et all. 2011. “System Analys and Desaign In A Changing World. Cengage Learning”, ISBN-13: 978-1111534158.